

**DEVELOPMENT AND VALIDATION OF A
MECHANISTIC, DYNAMIC MODEL FOR A VAPOR
COMPRESSION CENTRIFUGAL LIQUID CHILLER**

Sponsored by

ASHRAE

Deliverable for Research Project 1043-RP

Fault Detection and Diagnostic (FDD)

Requirements and Evaluation Tools for Chillers

HL 2002-8 Report #4036-4

Submitted by: Satyam Bendapudi, Graduate Research Assistant
 James E. Braun, Principal Investigator

Approved by: Robert J. Bernhard, Director
 Ray W. Herrick Laboratories

MAY 2002

TABLE OF CONTENTS

LIST OF FIGURES	iv
LIST OF TABLES	vi
NOMENCLATURE	vii
1 Introduction.....	1
2 Model Objectives	1
3 System Description	2
3.1 Instrumentation	4
3.2 Components	6
3.2.1 The Controller	6
3.2.2 Heat Exchangers	6
3.2.3 Compressor	8
3.2.4 Expansion valve	8
3.3 Data Collection	9
4 Model Description	12
4.1 Assumptions.....	13
4.2 Heat Exchangers	13
4.3 Heat Transfer Coefficients.....	18
4.4 Compressor	20
4.5 Expansion valve	23
4.6 Motor Cooling Line	24
4.7 Refrigerant Inventory.....	25
5 System Model Solution.....	26
5.1 Overview.....	26
5.2 Initialization	29

5.3	Compressor solution	30
5.4	Valve solution	35
5.5	Heat exchanger integration	35
6	Validation.....	39
6.1	Model settings.....	39
6.2	Fault-free condition.....	40
6.3	Reduced water flow rates.....	47
6.4	Refrigerant under-charge	52
6.5	Refrigerant overcharge.....	57
6.6	Condenser fouling.....	62
	CONCLUSIONS.....	67
	REFERENCES	69
	APPENDICES	73
	APPENDIX A: Linearization of the conservation equations	74
	APPENDIX B: Property evaluations	76
	APPENDIX C: Valve geometry approximation.....	78
	APPENDIX D: Model software user manual	80

LIST OF FIGURES

Fig. 1: Schematic of refrigerant flow paths	3
Fig. 2: Schematic of water flow paths	4
Fig. 3: Instrumentation on evaporator water circuit.....	5
Fig. 4: Instrumentation on condenser water circuit	5
Fig. 5: Constructional details of heat exchangers	7
Fig. 6: Schematic of expansion valve arrangement	9
Fig. 7: Overall information flow diagram in system model.....	12
Fig. 8: Heat exchanger discretization detail.....	14
Fig. 9: System simulation flow-chart.....	27
Fig. 10: Overall compressor model flow-chart.....	31
Fig. 11: Compressor exit enthalpy residual computation flow-chart.....	33
Fig. 12: Solution to the two-dimensional compressor quasi-steady-state model.....	34
Fig. 13: Information flow within heat-exchanger model	36
Fig. 14: Heat-exchanger integration algorithm.....	38
Fig. 15: Steady state model performance in fault-free condition.....	42
Fig. 16: Start-up model performance in fault-free condition.....	44
Fig. 17: Load change (LC9) model performance in fault-free condition	45
Fig. 18: Refrigerant charge during system simulation.....	46
Fig. 19: Steady-state model performance with 20% reduction in water flow-rates.....	49
Fig. 20: Start-up model performance with 20% reduction in water flow-rates	50
Fig. 21: Load change (LC9) model performance with 20% reduction in water flow-rates.....	51
Fig. 22: Steady-state model performance with 20% reduction in charge	54
Fig. 23: Start-up model performance with 20% reduction in refrigerant charge.....	55
Fig. 24: Load change (LC9) model performance with 20% reduction in refrigerant charge ...	56
Fig. 25: Steady-state model performance with 20% excess refrigerant.....	59
Fig. 26: Start-up model performance with 20% excess refrigerant	60
Fig. 27: Load change (LC9) model performance with 20% excess refrigerant.....	61

Fig. 28: Steady-state model performance with 45% fouling in condenser	64
Fig. 29: Start-up model performance with 45% fouling in condenser.....	65
Fig. 30: Load change (LC9) model performance with 45% fouling in condenser	66
Fig. 31: Approximated geometry used in valve model.....	78
Fig. 32: Screen-shot of unzipped chiller model files	80
Fig. 33: Screen-shot of the Geometries directory	81
Fig. 34: Heat exchanger geometry specification format.....	82
Fig. 35: Screen-shot of IOFiles directory	83
Fig. 36: Screen-shot of text file format for full initialization	84
Fig. 37: Screen-shot of text file format for minimal initialization.....	84
Fig. 38: Screen-shot of text file format for system states information	85
Fig. 39: m-code of Example 1.....	90
Fig. 40: Output plot of selected chiller parameters in Example 1.	91
Fig. 41: Output plot of selected chiller parameters in Example 2.	92
Fig. 42: m-code of Example 2.....	93
Fig. 43: Output plot of selected chiller parameters in Example 3.	95
Fig. 44: Output plot of selected chiller parameters in Example 4	96
Fig. 45: Output plot of selected chiller parameters in Example 5	97
Fig. 46: Output plot of selected chiller parameters in Example 6	98
Fig. 47: Screen-shot of condenser geometry file with 40% fouling introduced	99
Fig. 48: Screen-shot of error log file with an example error entry	102

LIST OF TABLES

Table 1: Steady-state conditions matrix (one test cycle)	11
Table 2: Constants and coefficients used in compressor model	22
Table 3: Surface enhancement correction factors	40
Table 4: Deviation of steady-states from fault-free with reduced flows	47
Table 5: Deviation of steady-states from fault-free with reduced charge.....	52
Table 6: Deviation of steady-states from fault-free with excess charge.....	57
Table 7: Deviation of steady-states from fault-free with fouled condenser ation.....	62
Table 8: Property tables and associated text files	76
Table 9: Valve parameters	79

NOMENCLATURE

Symbol	Description	Units
A_{cl}	Flow area of the cooling line orifice	m^2
$A_{in,i}$	Inside tube heat transfer area of the i^{th} node	m^2
$A_{out,i}$	Shell-side tube heat transfer area of the i^{th} node	m^2
A_v	Flow area through expansion valve	m^2
$A_{x,i}$	Cross-sectional flow-area of i^{th} node	m^2
α_c	Condensing heat transfer coefficient	$kW/m^2-^{\circ}C$
α_e	Evaporating heat transfer coefficient	$kW/m^2-^{\circ}C$
α_r	Refrigerant side heat transfer coefficient	$kW/m^2-^{\circ}C$
α_w	Water side heat transfer coefficient	$kW/m^2-^{\circ}C$
C_b	Bulb time constant	s
C_d	Coefficient of discharge	
$C_{p,t}$	Specific heat of tube material	$kJ/kg-^{\circ}C$
$C_{p,w}$	Specific heat of water	$kJ/kg-^{\circ}C$
$C_{sf,c}$	Refrigerant side surface enhancement correction factor for condensing region	
$C_{sf,sp}$	Refrigerant side surface enhancement correction factor for single phase region	
$C_{sf,we}$	Evaporator water side surface enhancement correction factor	
$C_{sf,wc}$	Condenser water side surface enhancement correction factor	
d_i	Inside diameter of tube	m
d_o	Outside diameter of tube	m
D_i	Inside diameter of shell	m
ΔP_{min}	Minimum pressure required to open valve	kPa
η_p	Polytropic efficiency	
η_{em}	Electro-mechanical efficiency	
f	Friction factor	
g	Acceleration due to gravity	m^2/s
γ	Normalized controller output	
h_1	Refrigerant enthalpy at evaporator exit	kJ/kg
h_2	Refrigerant enthalpy at compressor exit	kJ/kg

Symbol	Description	Units
h_3	Refrigerant enthalpy at condenser exit	kJ/kg
h_4	Refrigerant enthalpy at evaporator inlet	kJ/kg
h_{fg}	Enthalpy of vaporization	kJ/kg
h_i	Enthalpy of refrigerant in the i^{th} node	kJ/kg
h_{in}	Enthalpy of refrigerant entering the heat exchanger	kJ/kg
Ja	Jakob number	
k_l	Thermal conductivity of saturated liquid refrigerant	kW/m-°C
k_{spring}	Compliance of expansion valve spring	m/kPa
L	Tube length	m
\dot{m}_c	Refrigerant flow rate through compressor	kg/s
\dot{m}_{cl}	Refrigerant flow rate through cooling line	kg/s
$\dot{m}_{c,max}$	Wide-open vanes flow rate through compressor	kg/s
\dot{m}_i	Refrigerant mass flow rate leaving the i^{th} node and entering the $(i+1)^{\text{th}}$ node	kg/s
$\bar{\dot{m}}_i$	Mean refrigerant flow rate through node	kg/s
\dot{m}_{in}	Refrigerant flow rate entering heat exchanger	kg/s
\dot{m}_o	Refrigerant flow rate leaving the heat exchanger	kg/s
\dot{m}_v	Refrigerant flow rate through valve	kg/s
$\dot{m}_{w,i}$	Water flow rate in the i^{th} node	kg/s
$\dot{m}_{w,e}$	Water flow rate in evaporator loop	kg/s
$\dot{m}_{w,c}$	Water flow rate in condenser loop	kg/s
$M_{t,i}$	Mass of tube material in i^{th} node	kg
$M_{w,i}$	Mass of water in the i^{th} node	kg
μ_l	Viscosity of saturated liquid refrigerant	kg/m-s
N	Number of nodes	
$N_{T,i}$	Number of tubes in i^{th} node	

Symbol	Description	Units
Nu	Nusselt Number (subscripts: we – evaporator water, wc – condenser water, sp – single phase)	
P	Pressure (subscript: e – evaporator, c – condenser, b – bulb)	kPa
P_{motor}	Motor power	kW
Pr	Prandtl Number	
\dot{Q}_{cc}	Heat loss from motor and transmission	kW
$\dot{Q}_{r,i}$	Refrigerant heat flux in the i^{th} node	kW/m ²
$\dot{Q}_{r,i}$	Refrigerant side heat transfer rate in the i^{th} node	kW
$\dot{Q}_{w,c}$	Condenser water heat transfer rate	kW
$\dot{Q}_{w,e}$	Evaporator water heat transfer rate	kW
$\dot{Q}_{w,i}$	Water side heat transfer rate in i^{th} node	kW
RLA	Percentage of Rated Load Amps drawn by motor (subscript: max – limiting rated load amps)	%
Re	Reynolds number	
ρ	Density (subscripts: l – liquid, v – vapor, i - i^{th} node)	kg/m ³
T_1	Refrigerant temperature at evaporator exit	°C
T_b	Bulb temperature	°C
T_{cwo}	Condenser leaving water temperature	°C
T_{cwi}	Condenser entering water temperature	°C
T_{ewo}	Evaporator leaving water temperature	°C
T_{ewi}	Evaporator entering water temperature	°C
$T_{ewo,Set}$	Evaporator leaving water set point temperature	°C
$T_{r,i}$	Refrigerant temperature in the i^{th} node	°C
T_s	Tube surface temperature	°C
T_{sat}	Saturation temperature	°C
$T_{t,i}$	Tube temperature in the i^{th} node	°C

Symbol	Description	Units
$T_{w,i}$	Water temperature in the i^{th} node	$^{\circ}\text{C}$
$T_{w,in}$	Entering water temperature	$^{\circ}\text{C}$
$T_{w,o}$	Temperature of water leaving the heat exchanger	$^{\circ}\text{C}$
u	Specific internal energy	kJ/kg
v	Specific volume(subscript $i - i^{\text{th}}$ node)	m^3/kg
v_1	Refrigerant specific volume at evaporator exit	m^3/kg
v_2	Refrigerant specific volume at compressor exit	m^3/kg
v_3	Refrigerant specific volume at condenser exit	m^3/kg
V	Volume (subscript $i - i^{\text{th}}$ node)	m^3
\dot{V}	Volumetric flow rate of refrigerant	m^3/s
$\hat{V}_{r,i}$	Refrigerant flow velocity through i^{th} node	m/s
W_p	Specific polytropic work	kJ/kg
y	Expansion valve lift	m
y_i	Control volume length dimension along flow direction	m

1 Introduction

Dynamic models are crucial tools for the controls engineer in developing efficient control algorithms. Dynamic performance modeling of vapor compression systems has been of interest for well over 20 years, beginning with Dhar and Soedel [1979]. In preparation for this model development exercise, an extensive literature survey was carried out and is reported in a separate document (Bendapudi and Braun [2002]). Papers related to liquid chiller models include Sami et al [1987], Svensson [1999], Wang and Wang [2000], Browne and Bansal [2000] and Grace and Tassou [2000]. None of these models is comprehensive in that they either do not consider centrifugal compressors or they use simplified heat exchanger models that cannot adequately model large and small scale transients. Sami's model was limited to a hermetically sealed reciprocating compressor, while Browne's dealt with screw compressors only. Svensson's work focused only on transients triggered by feedback control. Wang's model, which does characterize a centrifugal liquid chiller model, utilizes very simple heat exchanger models. Grace and Tassou modeled a reciprocating compressor with a shell-tube evaporator that operated with refrigerant in-tube. The heat exchangers were modeled on the lines of MacArthur and Grald [1987]. Browne and Bansal [1998], in their compilation work on issues related to modeling of vapor compression liquid chillers, highlight the need for a liquid chiller model that incorporates detailed heat exchangers.

To summarize, it was found that no publicly available system models existed that could predict the complete dynamic performance of vapor compression centrifugal liquid chillers despite such systems being among the more popular configurations in the field.

2 Model Objectives

The objective of the work described in this report was to develop and validate a transient model of a vapor compression centrifugal liquid chiller system that:

- is based on first principles wherever available information permits;
- can capture start-up transients, as well as transients caused by changes in steady state operating points;
- can execute close to real-time, if not faster and;

- can be used to study the impact of common faults that occur in such systems;

In addition, the model developed was to be implemented in a form that would

- (a) allow execution from within a Matlab environment.
- (b) be modular in component models.

Based on these requirements, C++ was chosen as the coding language because of its object-oriented features and its ease of interface with Matlab.

3 System Description

Before beginning a description of the model developed, it is necessary to describe the system configuration, the test-stand details and data available for validation. As is inevitable in a project of this magnitude, the job was carried out as a team effort. The installation, instrumentation and commissioning of the test stand, data collection with and without faults introduced in the system and a detailed survey of important faults that occur in the field are documented in Comstock [1999]. The brief description of the system, associated figures and details of the test stand given below are extracted from the same and provided for the sake of completeness in the context of the dynamic model development. For further details about the test stand, instrumentation, faults implementation and data collection, please refer to Comstock [1999].

The equipment modeled is a McQuay PEH048J, 90-ton chiller. The system consists of a shell-and-tube evaporator, a shell-and-tube condenser, a pilot-driven expansion valve and a centrifugal compressor. Capacity control is achieved by varying the compressor's inlet-guide-vane angle. The refrigerant used in the system is R134a and the secondary fluid is water. A schematic of the system and refrigerant flow paths is shown in Fig. 1. Parallel to the liquid line that carries the bulk of high pressure refrigerant through the expansion device, there exists a small bypass line that is tapped at the exit of the condenser. This bypass line, hereafter called the cooling line, carries liquid refrigerant to the motor and transmission housing where it is first expanded across an orifice to the evaporator pressure, and then used to cool the motor and transmission oil. The refrigerant flows directly over the motor windings, but stays oil free as it cools the oil-cooler coil. The heated refrigerant is returned to the main refrigerant stream at the evaporator inlet.

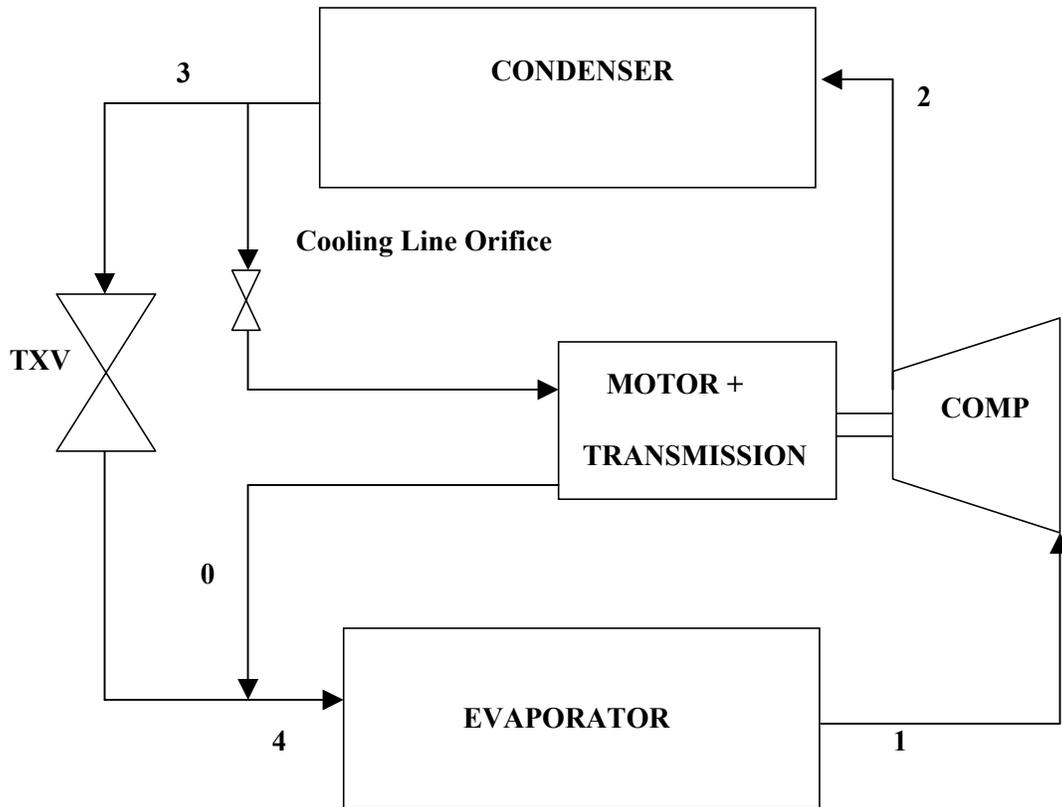


Fig. 1: Schematic of refrigerant flow paths

The test-stand schematic with water flow paths is shown in Fig. 2. The chiller test stand is provided with an arrangement of water-to-water and water-to-steam heat exchangers as the substitute for actual building and ambient loads. Chilled water from the evaporator passes through a shared heat exchanger where a part of the heat rejected from the condenser is transferred to the chilled water. Additional heat can be added to the evaporator water in the hot-water heat exchanger. The hot water for this heat exchanger is obtained using a steam heat exchanger. The temperature of the water entering the evaporator is adjusted by controlling the valves in the evaporator water circuit, the hot-water circuit and the steam circuit.

Similarly, hot water from the condenser first passes through a “city-water” heat exchanger where some of the heat is rejected to municipal water that runs in an open loop. Further heat rejection is achieved in the *shared* heat exchanger in which heat is transferred from the warm water in the condenser loop to the cooler water in the evaporator loop. Variation of the condenser water inlet temperature is achieved by controlling the valves in the

condenser water circuit, city-water line and a three-way valve that controls the fraction of condenser water that can bypass the shared heat exchanger.

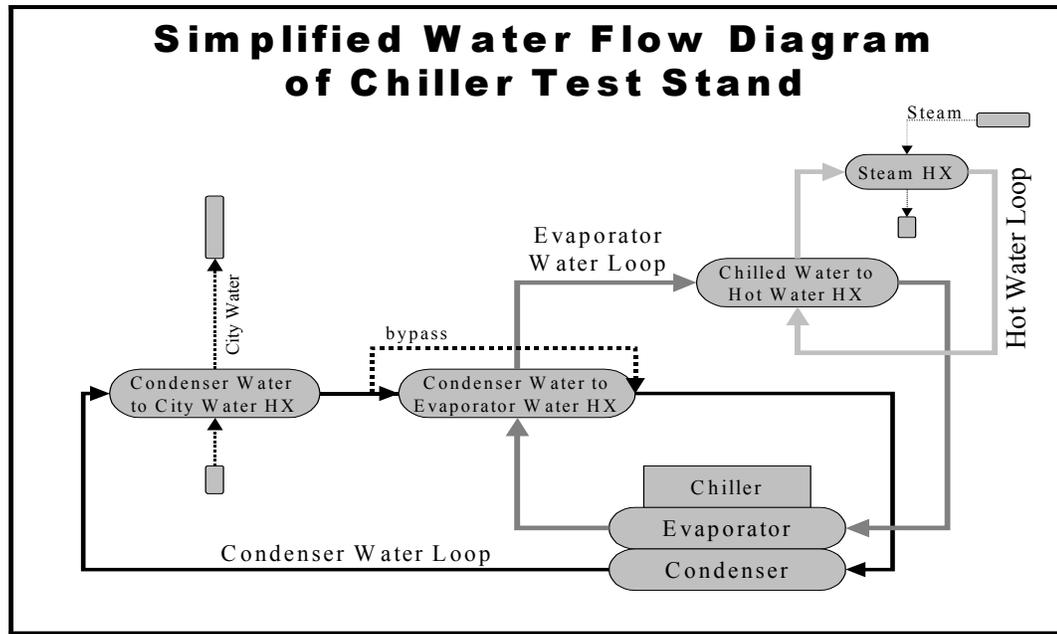


Fig. 2: Schematic of water flow paths

3.1 Instrumentation

The system is instrumented with temperature, flow and position sensors (for valve positions) in the water circuits, and pressure and temperature sensors in the refrigerant circuit. Locations of the sensors in the water circuits are shown in Figs. 3 & 4. For description of the nomenclature in these figures and sensor details, please refer to Comstock [1999]. In the refrigerant circuit, the temperatures and pressures at the compressor's inlet and outlet are measured. In addition to these, a power meter was installed at the motor to measure the electrical power consumption. All data from the system is communicated to a desktop PC running a VisSim interface, which is a Windows based simulation environment that is capable of handling real time data collection and analysis. Some of the measurements (from OEM sensors) are sampled by the chiller system's built-in controller while others are sampled and relayed to the PC through a bank of separate controllers, which control the positions of the valves in the water circuits. The VisSim interface provides an alternate interface with the chiller system and allows almost complete control of the system parameters.

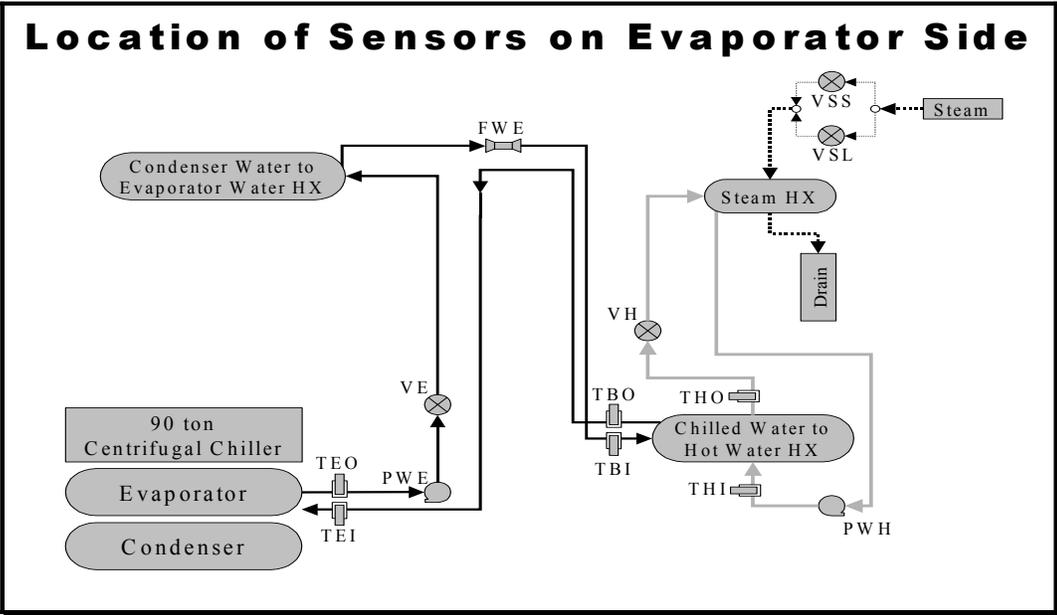


Fig. 3: Instrumentation on evaporator water circuit

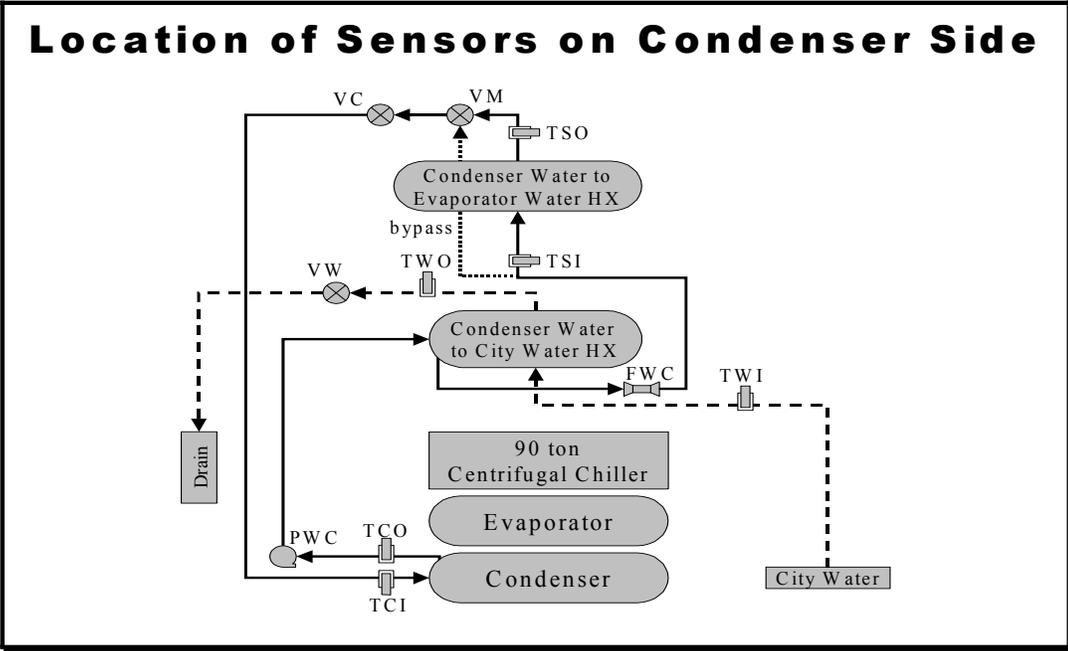


Fig. 4: Instrumentation on condenser water circuit

3.2 Components

3.2.1 The Controller

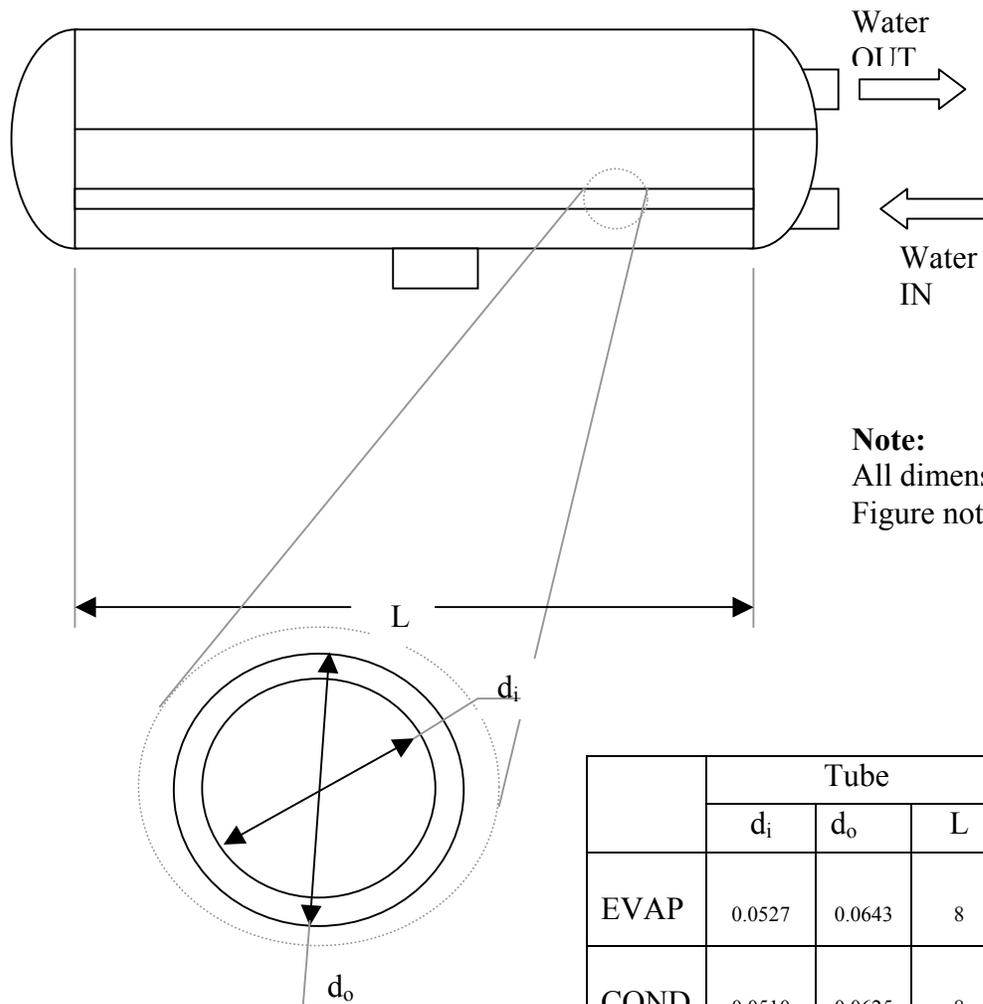
The system is overseen by a microprocessor based Microtech controller (hereafter referred to as the ‘controller’), which adjusts the compressor’s inlet guide vanes to maintain a specified water outlet temperature at the evaporator. The controller takes care of the following major tasks:

- Monitoring of system parameters for safe operation. This includes:
 - self-test at startup;
 - confirmation that all necessary systems have been switched on or off as required;
 - confirmation that alarms triggered from earlier runs have been cleared;
 - confirmation that all measurements are within tolerances acceptable for start-up;
 - appropriate shut-down sequencing in the event of unfavorable operating conditions such as abnormal system pressures or water flow rates.
- Output of appropriate control signals to the actuator of the compressors capacity control mechanism.
- Collection, storage and transmission of data through the communications channel.

3.2.2 Heat Exchangers

Both the evaporator and condenser are flooded-type, 2-pass shell-and-tube heat exchangers, with water as the secondary-coolant. The water flows in the tubes and the refrigerant flows outside them. The water enters in the lower half and leaves in the upper half. Refrigerant either enters at the top and leaves from the bottom, as in the condenser, or enters at the bottom and leaves from the top, as in the evaporator.

Fig. 5 shows the constructional details of the evaporator and condenser. All known quantities have been indicated in the figures. The tube-pattern in the condenser, as seen with the end cap of the shell removed, is shown as an example. The numbering of the tube rows follows the nominal flow-direction of refrigerant through the shell.



Note:
All dimensions in ft.
Figure not to scale

	Tube			Shell
	d_i	d_o	L	D_i
EVAP	0.0527	0.0643	8	1.289
COND	0.0510	0.0625	8	1.239

Tube row	No. of tubes	
	Evaporator	Condenser
1	3	13
2	6	14
3	10	15
4	12	16
5	13	16
6	14	16
7	16	16
8	16	14
9	16	13
10	16	12
11	14	10
12	13	6
13	-	3
<i>Total</i>	<i>149</i>	<i>164</i>

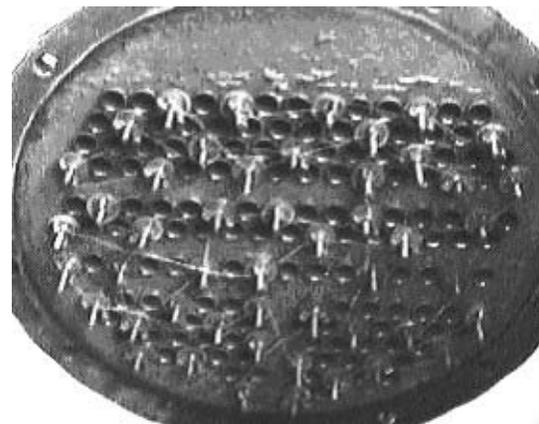


Fig. 5: Constructional details of heat exchangers

3.2.3 Compressor

The compressor is a centrifugal unit with a 4.8” impeller. It is driven by a constant-speed electric motor and transmission arrangement that keeps the impeller running at about 32,000 rpm. Capacity control is achieved by the opening and closing of the inlet guide vanes. The motion of the inlet guide vanes is controlled by means of an actuator connected to a double-acting piston. The piston moves within a cylinder under differential oil pressure across its faces. The vanes are closed or opened, depending upon which face of the piston is fed with oil at higher pressure. This, in turn, is controlled by an arrangement of solenoid valves that direct the high-pressure oil through the piston-cylinder. Opening (or closing) of the vanes requires a specific solenoid-valve to be opened. The extent of opening (or closing) is controlled by the time that the opening (or closing) solenoid-valve is energized / de-energized. This time is determined by the controller, and sent as a digital signal to operate the appropriate valve(s).

3.2.4 Expansion valve

The expansion arrangement consists of a main valve in the liquid line and a pilot-valve parallel to the liquid line. The pilot valve is a cross-charged thermostatic expansion valve (with R500 in the bulb) that provides the actuation of the main valve. In response to the superheat pressure, the pilot valve drops the condenser pressure to an intermediate pressure, which is then applied to the main valve piston. The main valve then opens or closes depending on the differential between the intermediate pressure and the evaporator pressure. Flow in the pilot-valve line is merged with the main valve flow after dropping the intermediate pressure down to the evaporator pressure, through a fixed-size orifice in the main valve’s piston. Fig. 6 shows the schematic of the expansion valve.

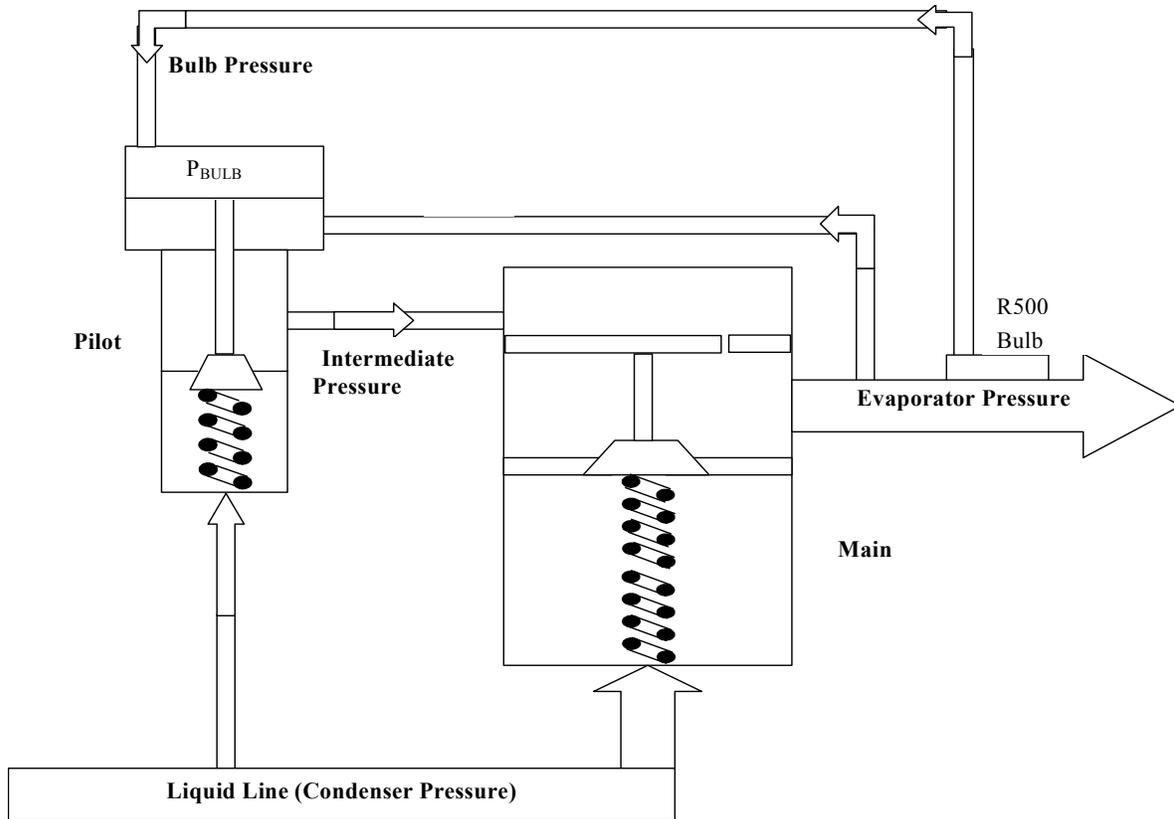


Fig. 6: Schematic of expansion valve arrangement

3.3 Data Collection

The data collection exercise consisted of pre-programming the chillers operation, through the VisSim interface, for 27 different combinations of capacity (evaporator water inlet temperature), ambient (condenser water inlet temperature) and set point (evaporator water outlet temperature) conditions. Table 1 presents these steady state conditions. After the test sequence was input, the system was run in a continuous sequence through all the 27 conditions and data was collected at 10s intervals, starting well before the compressor was switched on by the controller and ending shortly after the compressor was switched off. One complete run took a little over 14 hours and included start up and approach to the first steady-state condition followed by 26 conditions obtained by changes to one or more of the three independent variables mentioned above and shutdown. As can be seen in Table 1, the set point temperature changed least often. Keeping the chilled water set point temperature and the condenser water inlet temperature constant, the system was driven through a sequence of three steady-state

operating conditions between which the evaporator water inlet temperature was varied twice. This sequence of three was then repeated for two other condenser water inlet temperatures. This provided a sequence of nine operating conditions during which the set-point temperature remained constant. This sequence of nine was then repeated for two additional set-point temperatures.

Terming the above, along with the associated start-up, as one *test cycle* in the data collection process, multiple test cycles were run with different fault conditions implemented. The faults included reduced evaporator water flow, reduced condenser water flow, a combination of these two, condenser fouling, reduced system charge (refrigerant leakage) and increased system charge (refrigerant overcharge). Other faults implemented included excess oil in the compressor, a defective pilot valve and non-condensables in the refrigerant. Data was also collected with the system operating fault-free in order to benchmark the performance of the system.

Most of the above faults were run at different severity levels ranging from 10% to 40%. To summarize, each fault mentioned above was implemented at four different severity levels, and at each level of each fault, data was collected for the complete test cycle described above. The exception to this was the pilot valve fault, which was only tested at one severity level. Thus, the total number of transients over which data was available added up to about 760 at 27 transients per test cycle, one test cycle for each of 4 severity levels of 7 kinds of faults.

In order to ensure that the benchmark test cycle, i.e. no-fault condition, was not influenced in any way by the specific sequence of the test runs, the same test conditions were run in reverse order for the fault-free condition. For further details regarding the data collection process, along with the calibration tests, please see Comstock [1999].

For the purposes of brevity and clarity, the following terminology will be used in this document to refer to the various parts of the test cycle. “SS” followed by a number n , will refer to the test condition number in Table 1. “LC” followed by a number n , will refer to the transient associated with changing from steady-state condition number n to the next steady-state condition. For example, SS9 will refer to the steady-state condition number 9 and LC9 will refer to the transient behavior between SS9 and SS10.

Table 1: Steady-state conditions matrix (one test cycle)

Test No.	$T_{weo,Set}$ °F	T_{cwi} °F	T_{ewi} °F
1	50	85.6	60.6
2	50	85.28	57.07
3	50	85.12	53.6
4	50	75.48	61.27
5	50	74.9	56.58
6	50	75.15	52.68
7	50	69.98	58.45
8	50	65.34	55.95
9	50	62.52	53.49
10	45	85.27	56.16
11	45	85.3	51.3
12	45	84.95	49.18
13	45	75.23	55.95
14	45	75.21	50.21
15	45	75.03	48.34
16	45	70.23	55.5
17	45	64.96	51.28
18	45	64.76	48.52
19	40	80.11	50.0
20	40	80.03	46.48
21	40	80.11	44.08
22	40	70.16	51.84
23	40	69.26	45.55
24	40	68.25	43.55
25	40	63.36	48.64
26	40	61.14	45.88
27	40	60.83	43.02

4 Model Description

Fig. 7 presents a succinct overview of information flow within the system model and identifies the various input-output variables for each component. The solid arrows represent the refrigerant flow and the lighter arrows represent the information flow, the directions of the arrows indicating input or output. The symbols are defined within the nomenclature. States 1, 2, 3 and 4 are the refrigerant states at compressor inlet, condenser inlet, condenser outlet and evaporator inlet respectively.

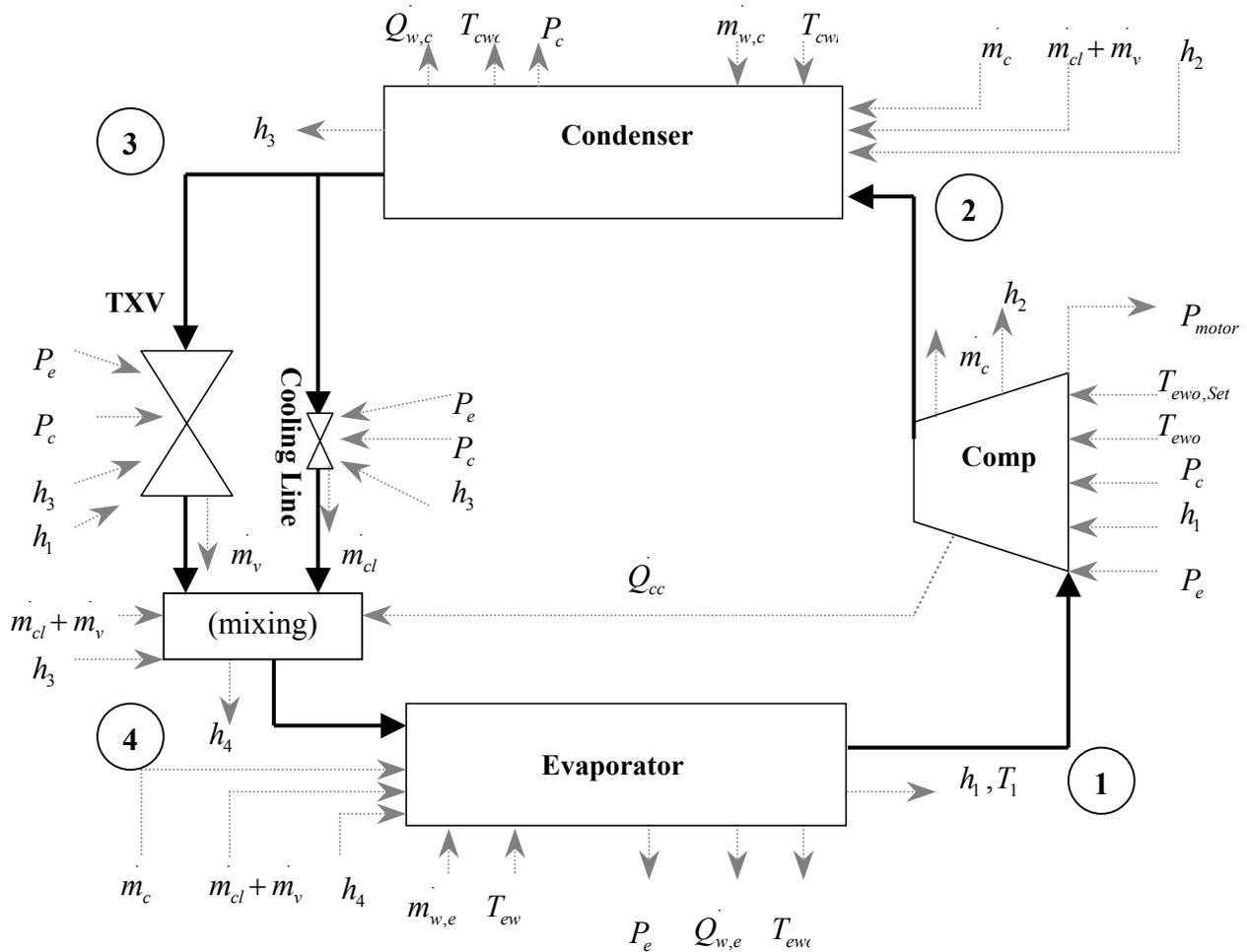


Fig. 7: Overall information flow diagram in system model

The following dynamics are considered in the model:

- refrigerant re-distribution between the heat-exchangers;
- thermal capacitance of the tube material;

- thermal capacitance of the water within the heat-exchanger;
- thermal capacitance of the valve's sensing bulb;
- programmed dynamics of the compressors guide-vane controller.

4.1 Assumptions

In constructing the system model, the following major assumptions were made:

- pressure drops in the heat exchangers and piping are negligible;
- expansion across the valve and the cooling line orifice is isenthalpic;
- compression is adiabatic;
- two-phase regions in the heat-exchanger are homogenous;
- water-flow is single-pass in both heat-exchangers;
- tube material conductance is infinite;
- shells are adiabatic;
- water in each node is fully mixed.

4.2 Heat Exchangers

The formulation used for refrigerant within the heat exchangers is identical to the one developed by Rossi and Braun [1999] to model the dynamics of the heat exchangers of a roof top unit. Since both the evaporator and condenser are governed by the same physics, it was decided to develop a generic heat exchanger model that could be specified to operate as a condenser or as an evaporator at the time of system execution. The only significant differences are in the correlations used for the heat-transfer coefficients. The common formulation is outlined in this section. Details specific to either heat exchanger are outlined in the subsequent sections on heat transfer coefficients. The heat exchanger model is a finite-volume, homogenous two-phase formulation with no pressure drop. The discretization of the refrigerant within the heat exchangers is shown in Fig. 8. The heat exchanger shell volume is divided into vertically stacked control volumes. Each control volume spans vertically, one or more rows of tubes and horizontally, the entire length of the tubes. The top and bottom horizontal faces of the control volumes coincide with the median plane between tube banks. The longitudinal end faces of the control volume coincide with the end plates of the heat exchanger. The lateral faces of the control volumes coincide with the inside (curved) face of the shell.

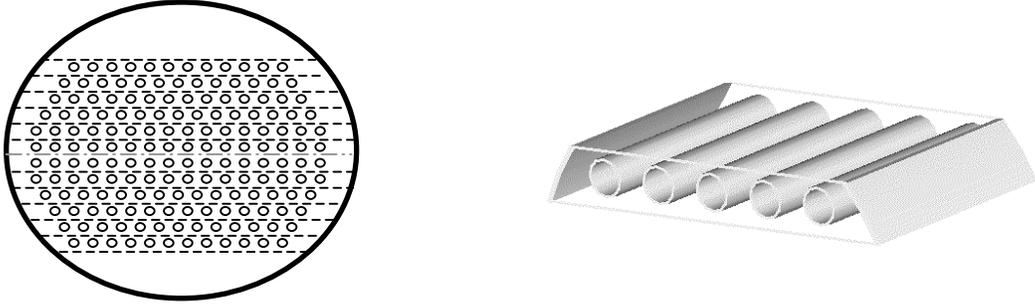


Fig. 8: Heat exchanger discretization detail

(sectional view of shell at left and perspective view of a typical control volume at right)

The relevant physical laws applicable to the phenomenon of refrigerant movement within the system are those that govern compressible fluid flow mechanics, and are, in the most general case, the laws of conservation of mass, energy and momentum. By the nature of the discretization chosen, the assumption of one-dimensional flow is enforced, and the appropriate conservation equations for refrigerant within a control volume i , are:

Mass conservation:

$$\frac{d(\rho V)_i}{dt} = \dot{m}_{i-1} - \dot{m}_i \quad (1)$$

Energy conservation:

$$\frac{d(\rho u V)_i}{dt} = \dot{m}_{i-1} h_{i-1} - \dot{m}_i h_i - \dot{Q}_{r,i} \quad (2)$$

where the convention adopted is of heat transfer leaving the control-volume being negative.

Since refrigerant pressure drop within the shell is negligible, the momentum-conservation equation vanishes.

After some algebraic manipulation, (please see Appendix A for details) the above equations can be reduced to the following:

$$a_i \frac{dP}{dt} + b_i \frac{dh_i}{dt} = \dot{m}_{i-1} - \dot{m}_i \quad (3)$$

$$c_i \frac{dP}{dt} + d_i \frac{dh_i}{dt} = \dot{m}_{i-1} h_{i-1} - \dot{m}_i h_i - \dot{Q}_{r,i} \quad (4)$$

where a , b , c & d are defined as:

$$a_i = V_i \left(\frac{\partial \rho}{\partial P} \right)_h \quad b_i = V_i \left(\frac{\partial \rho}{\partial h} \right)_P$$

$$c_i = V_i \left[h \left(\frac{\partial \rho}{\partial P} \right)_h - 1 \right] \quad d_i = V_i \left[h \left(\frac{\partial \rho}{\partial h} \right)_P + \rho \right]$$

For any node, eqns. (3) & (4) form a pair of coupled differential equations with the state variables of pressure and enthalpy. In solving these equations, a form of up-winding is enforced in that the refrigerant always flows in the same direction through every control volume in the heat exchangers, i.e. from the inlet to the outlet. For example, in the case of the condenser, where the orientation is such that refrigerant vapor enters at the top and liquid refrigerant leaves from the bottom, the refrigerant flow direction in any control volume is also from top to bottom. This allows for a node-numbering scheme that uses this *a priori* knowledge of the flow-direction. In reality however, pressure variations through the shell and other local effects could potentially lead to local flow-reversals and these would not be captured by the assumption made in this model. However, if the flow-directions enforced on the boundaries of the heat exchangers are consistent, the uniformity of pressure and the mass balance in each node, together ensure that all the intermediate flows are aligned in the nominal flow direction enforced by the boundary conditions.

Also, the compressor is modeled to predict a refrigerant flow rate that is always from the evaporator to the condenser. In fact, the only component that has the potential of driving refrigerant ‘backwards’ is the valve. This could happen under conditions of start-up when the system pressures fluctuate in a way that allows the condenser pressure to instantaneously fall below the evaporator pressure. Anticipating this possibility, the valve model was provided with a reverse-flow check valve. These modeling features, coupled with the uniformity of flow direction mentioned above, ensure that the refrigerant always flows in the same direction through all control volumes in either heat exchanger.

Eqns. (3) and (4) can be written for each of the nodes, and for the first ($i-1=0$) and last ($i =N$) nodes, the entering and leaving mass flow rates are known boundary conditions

imposed by the compressor and expansion valve. Also, the enthalpy entering the heat exchanger is known as the exit condition from either the compressor or the valve and cooling line.

Compiling the 2N conservation equations for a heat exchanger in matrix form gives a system of 2N equations in 2N unknowns that are: N enthalpy derivatives, the N-1 intermediate flow rates and 1 pressure derivative (eqn. (5)).

This system, $\mathbf{AX}=\mathbf{B}$, can be solved for \mathbf{X} , knowing the boundary conditions mentioned above, and the heat transfer rates between the refrigerant and the tube.

$$\mathbf{A} = \begin{bmatrix} a_1 & b_1 & 1 & 0 & \cdot & \cdot & \cdot & \cdot & \cdot & 0 \\ c_1 & d_1 & h_1 & 0 & \cdot & \cdot & \cdot & \cdot & \cdot & 0 \\ a_2 & 0 & -1 & b_2 & +1 & 0 & \cdot & \cdot & \cdot & 0 \\ c_3 & 0 & -h_1 & d_2 & h_2 & 0 & \cdot & \cdot & \cdot & 0 \\ \cdot & \cdot \\ \cdot & \cdot \\ a_{N-1} & \cdot & \cdot & \cdot & \cdot & \cdot & -1 & b_{N-1} & +1 & 0 \\ c_{N-1} & \cdot & \cdot & \cdot & \cdot & \cdot & -h_{N-2} & d_{N-1} & h_{N-1} & 0 \\ a_N & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 0 & -1 & b_N \\ c_N & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 0 & -h_{N-1} & d_N \end{bmatrix} \quad \mathbf{X} = \begin{bmatrix} \frac{dP}{dt} \\ \frac{dh_1}{dt} \\ m_1 \\ \frac{dh_2}{dt} \\ m_2 \\ \cdot \\ \cdot \\ \cdot \\ m_{N-1} \\ \frac{dh_N}{dt} \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} m_{in} \\ \left(m_{in} h_{in} + \dot{Q}_{r,1} \right) \\ 0 \\ \dot{Q}_{r,2} \\ 0 \\ \cdot \\ \cdot \\ \cdot \\ -m_o \\ \left(-m_o h_N + \dot{Q}_{r,N} \right) \end{bmatrix} \quad (5)$$

The nodal heat transfers used in the input vector \mathbf{B} are determined using the rate equation:

$$\dot{Q}_{r,i} = \alpha_r A_{out,i} (T_{r,i} - T_{t,i}) \quad (6)$$

Eqn (6) captures the heat transfer only between the refrigerant and the tube-wall. In reality, the refrigerant also exchanges heat with the shell. There are two aspects of this heat transfer. One is the transient effect, where-in the shell acts as a sink (or a source) when the refrigerant pressure changes. This effect applies to both heat-exchangers and comes into play during pressure transients. The impact of this is to dampen the rate of pressure change in the shell. The second effect is at steady state, when heat is exchanged between the refrigerant inside the shell and the ambient air outside the shell. This is a negligible effect since the heat transfer from the shell is governed by natural convection, while that inside is due to

evaporation or condensation. Also, since the evaporator is normally insulated on the outside against such losses, the issue is pertinent only to the condenser, and can be safely neglected.

Referring to Fig. 8, the discretization scheme treats all the tube material in one or more tube-rows as a single lump. The temperature of the tube is determined from an energy balance across the total tube material in that node, as:

$$M_{t,i} C_{p,t} \frac{dT_{t,i}}{dt} = \left(\dot{Q}_{r,i} - \dot{Q}_{w,i} \right) \quad (7)$$

The water-side heat transfer rate is determined as

$$\dot{Q}_{w,i} = \alpha_w A_{in,i} (T_{t,i} - T_{w,i}) \quad (8)$$

As with the tube material, all the water within the tubes in the node is treated as a single, fully-mixed lump and the nodal water temperature is determined from an energy balance on the total water resident in that node, as:

$$M_{w,i} C_{p,w} \frac{dT_{w,i}}{dt} = \dot{m}_{w,i} C_{p,w} (T_{w,in} - T_{w,i}) + \dot{Q}_{w,i} \quad (9)$$

This model captures a water temperature profile along the flow direction of the refrigerant, i.e. from one tube row to the next. It cannot however, capture the variation in tube or water temperature along the length of each tube.

The temperature of the water leaving the heat exchanger is determined from an energy balance on nodal water temperatures, as:

$$T_{w,o} = \frac{\sum_{i=1}^{N_N} (T_{w,i} N_{T,i})}{\sum_{i=1}^{N_N} N_{T,i}} \quad (10)$$

As seen in Fig. 7, the heat exchanger models require the refrigerant flow-rates at the inlet and outlet, the refrigerant enthalpy at the inlet and the water temperature and flow-rate, and compute the refrigerant pressure, enthalpy distribution and leaving water temperature as outputs.

4.3 Heat Transfer Coefficients

Refrigerant

The refrigerant heat transfer coefficient used in eqn. (6) depends on the phase of the refrigerant. In two-phase heat transfer, it also depends on the direction of heat transfer, i.e. condensation or boiling. In reality, condensation and boiling within the shell are complicated processes with different liquid accumulation patterns around the tubes. Within the shell, from entry to exit, the refrigerant goes through a variety of heat-transfer regimes. In order to simplify computation of the heat transfer coefficients, only four major regimes were considered for this model, namely, superheated, sub-cooled, condensing and evaporating. The boiling heat transfer coefficient is calculated from a correlation supplied by the chiller manufacturer, as

$$\alpha_e = C_1 + \dot{Q}_{r,i}'' / C_2 \quad (11)$$

with the boiling heat flux in kW/m², and α_e in kW/m²-K and C_1 and C_2 being manufacturer provided constants that are specific to the evaporator.

The condensing heat transfer coefficient is calculated using a correlation developed by a Nusselt laminar condensation analysis (Nusselt [1916]) applied to radial systems (Dhir and Lienhard [1971]):

$$\alpha_c = C_{sf,c} \cdot 0.729 \left[\frac{g \rho_l (\rho_l - \rho_v) k_l^3 h'_{fg}}{\mu_l (T_{sat} - T_s) d_o} \right]^{1/4} \quad (12)$$

where

$$h'_{fg} = (1 + 0.68Ja) h_{fg} \quad (13)$$

The refrigerant heat transfer coefficients in the superheated region in the evaporator and sub-cooled region in the condenser are calculated from a correlation of the form:

$$Nu_{sp} = C_{sf,sp} \cdot 1.13 \sqrt{\text{Re Pr}} \quad (14)$$

The refrigerant flow velocities required for computing the Reynolds numbers were determined using a mean refrigerant flow through the node and a flow area computed as:

$$\dot{\bar{m}}_i = \left(\dot{m}_{i-1} + \dot{m}_i \right) / 2 \quad (15)$$

$$\hat{V}_{r,i} = \dot{\bar{m}}_i / (\rho_i A_{x,i}) \quad (16)$$

$$A_{x,i} = V_i/y_i \quad (17)$$

The y_i 's are the vertical distances between the refrigerant inlet and outlet faces of the control volumes, and are input as part of the geometry of the heat exchanger. These were computed from the known tube arrangements within the heat exchangers.

The heat transfer coefficients computed using the above correlations were tuned during overall system simulation. The correction factors account for unknown tube-surface enhancements such as fins, rifling etc. Further detail is provided in Section 6 where the validation is described.

The two-phase heat transfer coefficients described above are inherently intended for use when the refrigerant quality is reasonably away from vapor and liquid states. As such, the two-phase correlations cannot be applied to nodes where the quality is very low or very high. Using these correlations for such nodes results in artificially large changes in heat transfer rates during the transition between single and two-phase, because the heat transfer coefficients in two-phase are orders of magnitude higher than those in single phase. These large changes make the solution of the state equations very difficult. To overcome this problem, the following approach was adopted to estimate reasonable heat transfer coefficients for such nodes. The two-phase correlations are applied only for nodes that are between 5% and 95% quality. When the node quality goes out of this range, the heat transfer coefficient is linearly interpolated between the two-phase value and the appropriate single-phase value. This approach makes for a smooth transition in the heat transfer coefficient, and eliminates numerical problems.

Water

The water-side heat transfer coefficients for the evaporator and condenser are calculated from standard correlations available in the literature:

$$Nu_{we} = C_{sf,we} \cdot \frac{(f/8)(Re-1000)Pr}{1+12.7(f/8)^{1/2}(Pr^{2/3}-1)} \quad (\text{Gnielinski [1976]}) \quad (18)$$

$$Nu_{wc} = C_{sf,wc} \frac{(f/8)RePr}{1.07+12.7(f/8)^{1/2}(Pr^{2/3}-1)} \quad (\text{Petukhov [1970]}) \quad (19)$$

The friction factors were computed from correlations provided by McQuay. When computing the Reynolds number for the water, the flow-velocity must account for the reduced

water-flow velocity caused by the single-water pass assumption. For example, if both passes have equal number of tubes, doubling the computed single-pass flow-velocity (only for the purpose of computing the Reynolds number) will suffice.

4.4 Compressor

The compressor is a centrifugal machine with inlet guide vane capacity control. The controller's actuator and the compressor are housed in an integral casing that precludes any direct measurement of the vane-position. Introducing instrumentation to measure the vane position for the purpose of this project would require significant, intrusive modifications to the compressor housing.

The compressor and the controller are modeled in three parts: the controller, a maximum capacity map to represent the behavior of the compressor if operating with wide open vanes, and a quasi-steady-state relationship between compressor power and flow with vanes partially closed. The operation of the controller is by a simple step-and-wait approach. The controller samples the chilled water temperature at fixed intervals and the difference from the set-point chilled water temperature is computed. Based on the magnitude and sign of this difference, the required control action is computed. The control action in the physical system is the incremental opening or closing of the inlet guide vanes. In the model, this is implemented by varying a factor γ (bounded between 0.05 and 1.0) that corresponds to a normalized vane-position. γ is increased or decreased incrementally depending on whether the compressor needs to be loaded or unloaded. Once the required control action has been determined, the actuator is driven, i.e. the vanes are opened (γ is increased) or closed (γ is decreased) at a known and finite rate. This actuator response speed was provided by the manufacturer.

A maximum flow-rate is computed for the compressor, assuming that it is operating under the given boundary conditions and with wide-open vanes. The actual flow rate is computed by applying the controller's vane-position correction to this maximum flow. The corrected flow rate is used in the quasi-steady state model to estimate the exit condition and power.

The quasi-steady-state model of the compressor was built from steady-state data obtained from the test stand. This model takes as inputs the inlet pressure and enthalpy and the exit pressure and flow-rate. The model outputs are the exit enthalpy, power requirement and motor losses.

From steady-state data, polytropic work, polytropic efficiency and heat loss in the motor-transmission were estimated. The polytropic efficiency was then regressed as a 2nd order polynomial in the volumetric flow rate (in m³/s) and polytropic work (in kJ/kg) as follows:

$$\eta_p = a_0 + a_1 \dot{V} + a_2 \dot{V}^2 + a_3 W_p + a_4 W_p^2 \quad (20)$$

Table 2 lists the coefficients used in this equation.

The polytropic compression work is given by:

$$W_p = (P_c v_2 - P_e v_1) \frac{\text{Ln}[P_c/P_e]}{\text{Ln}[(P_c v_2)/(P_e v_1)]} \quad (21)$$

Also, from the steady-state data, under the assumption of adiabatic compression, a constant electro-mechanical efficiency of the motor and transmission was computed. From this information, the power drawn by the motor and motor losses are calculated as:

$$P_{motor} = \dot{m}_c \frac{W_p}{\eta_p \eta_{em}} \quad (22)$$

$$\dot{Q}_{cc} = (1 - \eta_{em}) P_{motor} \quad (23)$$

Finally, an energy balance across the compressor gives the exit enthalpy, from:

$$P_{motor} = \frac{\dot{m}_c (h_2 - h_1)}{\eta_{em}} \quad (24)$$

In order to compute a maximum capacity flow, a physical model for the compressor was built from available geometry of the impeller using the formulation developed by Braun et al [1987] for variable-speed centrifugal compressors. This model was used to build a 2nd order regression model that computes the maximum capacity flow rate, in a “wide-open-vane” condition:

$$\dot{m}_{c,\max} = c_0 + c_1 P_e + c_2 P_c + c_3 T_1 + c_4 P_e P_c \quad (25)$$

Table 2 lists the coefficients used in this equation for pressures in kPa and temperatures in Celsius. The mass flow rate is in kg/s. The actual mass flow rate through the compressor is then calculated as:

$$\dot{m}_c = \gamma \cdot \dot{m}_{c,\max} \quad (26)$$

Within the control algorithm, there exists logic that prevents the compressor from loading if the current drawn by the motor exceeds the rated amperage of the motor. This feature is incorporated into the model by monitoring the amperage predicted by the model, as a percentage of the rated amps of the motor. This quantity is referred to as the ‘RLA’, and was regressed from the predicted power as:

$$RLA = b_0 + b_1 P_{motor} + P_{motor}^2 \quad (27)$$

Table 2 lists the coefficients used in this equation for motor power in kW and RLA as the percentage of full load amps. The current drawn by the motor (which is a 3-phase machine) is governed by Ohms law and is supposed to be proportional to the power if the power factor is constant. The zero-offset and the 2nd order term were found to be necessary to account for non-linearities in the system such as changes in the phase factor and hysteresis.

Table 2: Constants and coefficients used in compressor model

Polytropic Efficiency		Rated Load Amps		Wide-open flow rate	
a_0	-0.26524	b_0	7.2058	c_0	1.4354
a_1	7.1149	b_1	0.8	c_1	0.0054572
a_2	-23.415	b_2	0.003	c_2	-0.0030135
a_3	0.04173	Electro-mechanical efficiency		c_3	-0.017697
a_4	-0.00089576	η_{em}	0.75	c_4	0.000005653

When the current is below the limit, the compressor's quasi-steady-state model is solved with flow-rate (computed by the controller) as an input, and the power is an output.

When the limit is reached, the quasi-steady-state model is turned around, and the current limit is imposed as an input and the required mass flow rate and exit enthalpy are computed as outputs. This former solution of the model is accomplished by using the Secant method to converge on a residual of the exit enthalpy. The latter solution of the model involves a two-dimensional solution in the two unknowns of exit enthalpy and power. This is done using Newton's method with damping.

Overall, the compressor model requires the evaporator pressure, inlet enthalpy, condenser pressure and chilled water temperature error as inputs and outputs the refrigerant enthalpy at compressor discharge, refrigerant flow-rate, motor power and electro-mechanical losses.

4.5 Expansion valve

Constructing a physical model of the pilot-main valve system was complicated by the unavailability of either internal physical details or performance maps. Therefore, a simplified model of a standard thermostatic expansion valve was built and used in the system model. The simplified model incorporates a lumped-capacitance element to represent the bulb. The bulb, charged with R500, supplies the superheat pressure that acts against the evaporator pressure, to balance the valve opening. The bulb is modeled using:

$$C_b \frac{dT_b}{dt} = (T_1 - T_b) \quad (28)$$

The term C_b incorporates the effects of the refrigerant mass in the bulb, the thermal-mass of the bulb body, as well as the heat-transfer resistance on both sides of the bulb body. It may be viewed as an overall impedance term and also as a time-constant of the bulb. Determination of the exact time-constant of the bulb requires either a complete physical description of the bulb body and the refrigerant inside or frequency response data that isolates the bulbs response from that of the system complete. In the absence of either, C_b was estimated from the data using the superheat response time during load changes triggered solely by changes in either the condenser or the evaporator entering water temperatures. It was found that a change in $\sim 2^\circ\text{C}$ in either of these temperatures resulted in a superheat transient lasting

~200s. The bulb time constant was therefore set as 100s. Integrating eqn. (28) gives the bulb-temperature, and therefore the bulb pressure. The lift of the valve is computed as:

$$y = k_{spring} (P_b - P_e - \Delta P_{min}) \quad (29)$$

where ΔP_{min} is the minimum pressure difference required to open the valve, i.e. the start-open superheat pressure. The lift is used to compute the flow area, as:

$$A_v = a_0 y + a_1 y^2 \quad (30)$$

The constants a_0 and a_1 are determined from an assumed orifice profile, detailed in Appendix C. The flow-area is bounded between 0 and a maximum value. This ensures that the lift cannot be negative, or exceed a maximum limit. The spring stiffness and opening superheat were tuned during system simulation to achieve the measured superheat. Knowing the flow area of the valve, the flow rate can be computed from the orifice equation:

$$\dot{m}_v = C_d A_v \sqrt{2(P_c - P_e)/v_3} \quad (31)$$

where v_3 is the refrigerant specific volume at the condenser outlet. The discharge coefficient C_d was taken to be 0.4. A precise value is not crucial as it appears in eqn. (31) as a product with A_v which itself is determined using approximated coefficients (see Appendix C). Changes in C_d can be viewed as simply scaling a_0 and a_1 .

The inputs for the valve are therefore the high and low side pressures and exit enthalpies from the evaporator and the condenser. Using these, the valve model determines the flow-rate through the valve.

4.6 Motor Cooling Line

In addition to the above components, a part of the refrigerant flows in a parallel path through the compressor's motor shell for cooling the motor core and the transmission oil. This refrigerant is tapped at the condenser outlet, fed through the motor body and the oil cooler and returned to the evaporator shell after the main expansion valve. In order to balance the pressures, there exists an orifice in the motor cooling line that drops the pressure of the refrigerant from the condenser pressure to the evaporator pressure. Since the compressor is assumed adiabatic, all losses from the motor are treated as being added to the liquid line. The refrigerant enthalpy at the evaporator inlet can be determined by:

$$h_4 = h_3 + \dot{Q}_{cc} / (\dot{m}_v + \dot{m}_{cl}) \quad (32)$$

The flow through the cooling line is modeled as flow through an orifice of constant size, the diameter being provided by the manufacturer:

$$\dot{m}_{cl} = C_d A_{cl} \sqrt{2(P_c - P_e) / v_3} \quad (33)$$

The discharge coefficient used in eqn. (33) was taken to be 0.4. Early system simulations used coefficients of ~0.8. With these values it was found that when the valve would close, (as it does when the superheat falls below the minimum required to open it) the refrigerant fed to the evaporator through the cooling line would prevent the superheat from rising sufficiently to re-open the valve, thereby interfering with the valve's ability to control superheat.

4.7 Refrigerant Inventory

The system model built by combining the component models developed in the preceding sections is incomplete without the specification of the total quantity of refrigerant in the system. The total charge in the test-stand is known to be 136.1 kg. This refrigerant resides primarily in the evaporator, the condenser and the liquid line. The other refrigerant lines in the system almost always carry vapor refrigerant and contribute little to the total charge account. Since the liquid line is not modeled in the current system model, the overall charge quantity used in the model needs to be adjusted to discount the (essentially constant) refrigerant quantity that resides in the liquid line. Preliminary system simulations using this reduced charge were found to over-predict the condenser pressure and sub-cooling as would be expected if the system were overcharged. The total charge quantity to be used for the system model was tuned to a final value of 124.7 kg, by matching the predicted sub-cooling with the measurement in the fault-free condition. An explanation of the necessity for this additional adjustment of total refrigerant charge follows.

The total system charge is the sum of refrigerant quantities in the condenser and the evaporator. Within each heat exchanger, the total charge is the sum of refrigerant masses in each node. In the single-phase nodes, this can be easily determined from the volume of the node and the state of the refrigerant in that node. In two-phase nodes, an assumption is

required regarding the distribution of refrigerant through the volume of the node. In the two-phase regions of the condenser, condensed liquid refrigerant continually collects around the tubes and flows over the tubes vertically below. At any instant in time, all tubes in the two-phase region are surrounded by a layer of liquid refrigerant, and the remaining volume in the two-phase region of the shell is occupied by saturated vapor refrigerant. The thickness of the liquid layer around any tube depends on the rate of condensation on that tube as well as on the tubes above. An accurate inventory of refrigerant in the condenser needs to account for this non-uniform distribution of refrigerant within the two-phase region. The assumption of homogeneity in the two-phase regions bypasses this necessity by assuming that the entire refrigerant volume within any two-phase node is at a uniform quality. This allows the determination of a uniform density for the node. However, the homogenous model of two-phase results in an under-prediction of the refrigerant charge (Wallis [1969]) as compared to a separated flow model that accounts for the slip between the vapor and liquid phases. A similar issue occurs in the evaporator also, but since the bulk of the refrigerant is in the condenser, during operation, the phenomenon is more significant there. This difference is believed to be in some part responsible for the necessity to correct the total charge used in the system model.

5 System Model Solution

5.1 Overview

Referring to Fig. 9, the program execution begins with the loading of property data for R134a into memory followed by the reading of geometric and other system-specific information from text files. This is followed by an initialization of the system, i.e. setting values for the various state variables in the system as described below in section 5.2. The numerical solution of the system model is structured as three nested loops each characterized by a different time scale.

The outermost loop runs the simulation in 10s^1 steps. Every 10s, this loop updates the external boundary conditions which are the water flow rates and entering temperatures in the

¹ The 10s step size is chosen here to match the sampling rate of the measured data and is not enforced in the code. The outer loop can be driven through any positive integer number of seconds. When running simulations through the Matlab interface developed for this model, the user is required to specify this step size along with the external boundary conditions.

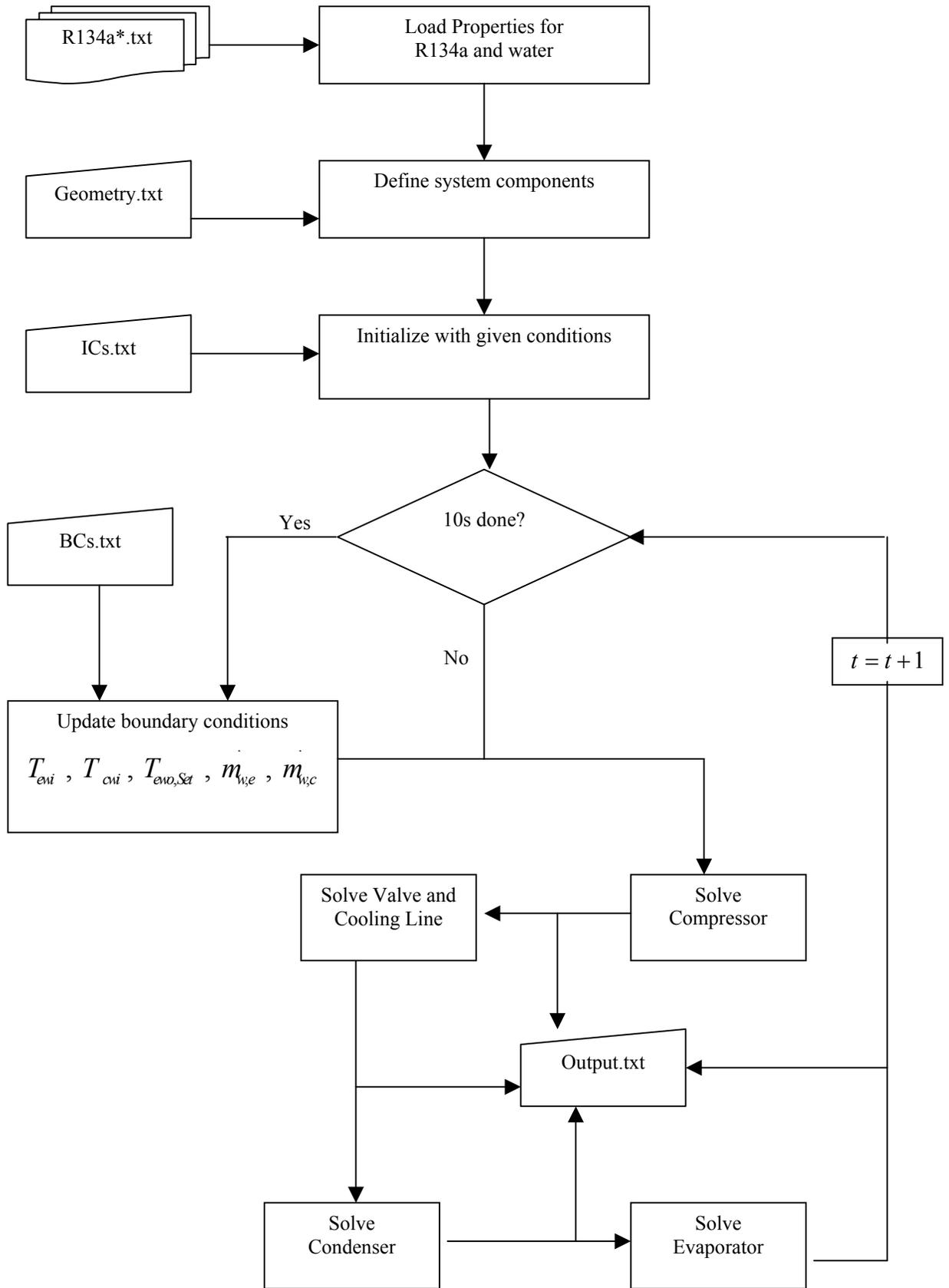


Fig. 9: System simulation flow-chart

condenser and evaporator loops and the chilled water set point temperature. Between successive updates these conditions are maintained constant for the inner loops.

The second level of the nested solution loop executes at 1s steps. In this loop, the system components are integrated forward in time using an explicit algorithm. Detailed solution of each of the components is described in subsequent sections. The following serves as a quick overview of this loop.

First to be solved is the compressor model. From current conditions of the compressor model inputs described in Section 4.4, the actuator position (γ) is increased or decreased. At this new position, the new compressor flow-rate is computed and other compressor outputs are determined.

Next to be solved are the valve equations, which provide the updated valve flow-rate using current evaporator and condenser pressures and exit enthalpies. This is followed by the cooling line model which computes the enthalpy at the evaporator inlet using the electro-mechanical losses determined from the compressor model. The condenser equations are then solved using the updated compressor and valve flow-rates and compressor exit enthalpy. The water flow-rate and entering temperature are known from the outermost loop. The condenser equations are integrated 1s forward in time explicitly, using an Euler one-step correction method described later (section 5.5). This determines the condenser pressure and enthalpy distribution at the end of 1s, which is used in the next iteration by the valve and compressor models.

Finally, the evaporator equations are solved exactly like the condenser and the updated evaporator pressure and enthalpy distribution are determined. These are used in the next iteration by the valve and compressor models. This completes one loop through the second level, and the system cycle starts again with the compressor model. As is seen from this, the internal boundary conditions required by each of the components are updated at 1s intervals and are maintained constant during these intervals.

The third and innermost loops correspond to the explicit Euler integration of the differential equations in the heat exchangers and the valve. The valve, the condenser and the evaporator models each execute their own independent integration loops with their own time-steps. More detail on the computations within this loop is deferred to section 5.5 which describes the integration algorithm.

5.2 Initialization

The preceding description of the model requires the specification of an initial condition in order for the problem to be well-posed and solvable. The initialization of the system consists of setting values for evaporator and condenser pressure, refrigerant enthalpies, tube and water temperatures of all nodes in both heat exchangers, specifying the initial control factor value and whether or not it is in start-up mode² and finally, specifying the temperature of the bulb.

Normally, when the compressor has not been switched on for an extended period of time, the refrigerant migrates to the coldest part of the system. This is usually the evaporator because it is insulated and has cold water running through the tubes. Eventually, the system attains equilibrium when the refrigerant pressures in both heat exchangers equalize. At this time, the condenser is normally filled with superheated vapor at the temperature of the water flowing through the condenser and at a pressure that corresponds to the saturation pressure of the temperature of the water flowing through the evaporator, which is filled with two-phase refrigerant. It is possible that the condenser could contain two-phase refrigerant as well if the condenser water is sufficiently cold.

Using the evaporator water leaving temperature, the system pressure is computed as the refrigerant saturation pressure corresponding to this temperature. At this pressure and at the temperature of the condenser leaving water, the superheated enthalpy of refrigerant in the condenser is determined. This enthalpy is assumed uniform throughout the shell volume of the condenser and the mass of refrigerant therein is determined. The remaining refrigerant (found by using the normal system charge determined as in Section 4.7 or a deviation from this normal system charge if implementing a fault such as under or over charging) is assumed to be in the evaporator. Knowing the shell-volume of evaporator and the mass of refrigerant, the specific volume can be determined. Since the pressure is uniform in the system, this uniquely identifies the quality and therefore the enthalpy of refrigerant in the evaporator. This enthalpy is assumed to be uniform throughout the evaporator shell.

² For the purposes of this model, the system is always assumed to begin from an internal and external equilibrium condition and to begin operating from start-up. Therefore, the control factor would always begin at a value of 0.05, which corresponds to the vanes being in the minimum-area position.

The bulb temperature is also initialized to the temperature corresponding to the evaporator since it is in thermal equilibrium with the refrigerant.

5.3 Compressor solution

As mentioned in Section 5.1, the compressor equations are the first to be solved in each loop through the system components. Referring to Fig. 10, from values of chilled water set point temperature, the current evaporator water outlet temperature and the power, the control factor is calculated using the manufacturer's proprietary controller algorithm. All the dynamics in the compressor model are encapsulated in this algorithm. The output of this algorithm is the control factor γ . From current evaporator and condenser pressures, and the evaporator exit condition, the maximum capacity flow rate is computed using eqn (25). The compressor mass flow rate is then calculated from the control factor and the just computed maximum capacity flow rate, using eqn (26). The control algorithm also computes the maximum allowable RLA, which sets the limit for how much power the motor is allowed to draw. This maximum limit follows a ramp profile during the first 10 minutes of start up and reaches (and remains at) 105% thereafter.

From current inlet conditions to the compressor, condenser pressure and mass flow rate, the quasi-steady-state compressor model is used to determine the new exit condition from the compressor, the motor power (and the corresponding RLA) and the heat losses. This computed RLA needs to be compared with the limiting RLA set by the controller to see if the computed flow-rate and power are permissible. If the RLA computed is below the limit, the compressor solution is acceptable and the program control moves to the valve. If however, the RLA exceeds the limit, the power corresponding to the limiting RLA is calculated from eqn. (27). This limiting power is used as an input to the quasi-steady-state model instead of the flow-rate, and the exit enthalpy and flow-rate that satisfy this power constraint are determined, all other boundary conditions remaining unchanged.

The solution for the exit enthalpy when the flow rate is known involves a single iteration variable. When the power is the known quantity instead of the mass flow rate, the solution involves a two-variable search. This is the reason for the terminology of "1D QSS

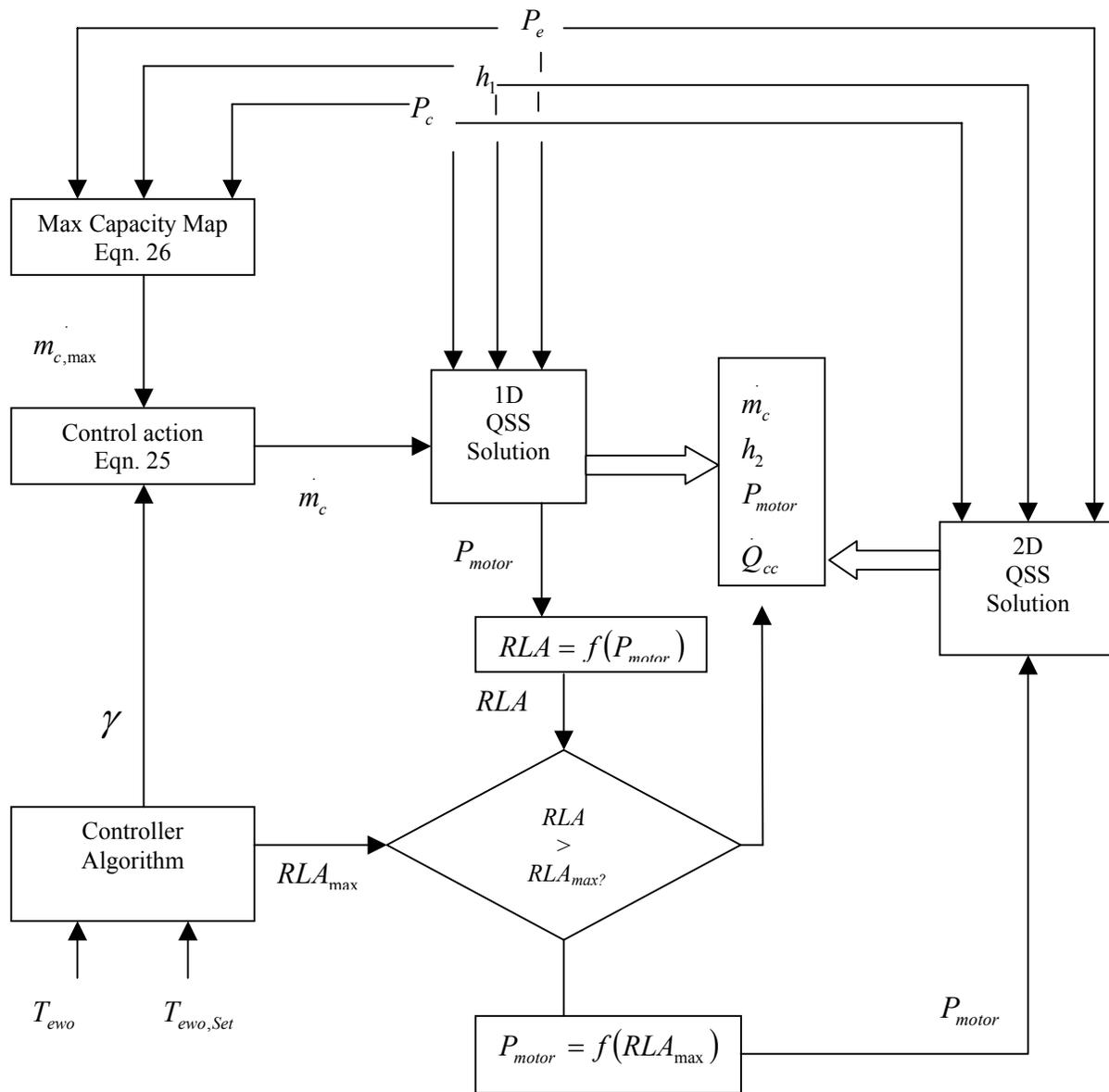


Fig. 10: Overall compressor model flow-chart

Solution” and “2D QS Solution” in Fig. 10. In either case, at the end of the compressor model solution, the exit enthalpy, motor power, flow-rate and heat losses are output.

The solution to the quasi-steady-state model when the mass flow rate is known (i.e. the 1D QSS Solution) is shown in Fig. 11. Beginning with a guess value for the exit enthalpy, the polytropic work is computed using eqn (21). From the compressor mass flow rate and the inlet state of the refrigerant, the volumetric flow rate is calculated. The polytropic work and volumetric flow rate are used to compute the polytropic efficiency from eqn (20). Care should be taken to ensure that the polytropic efficiency calculated is indeed physically meaningful. Negative or high (>1) values would indicate extrapolation. In order to avoid this, the polytropic efficiency computed using eqn. (20) is checked to see if it is between 0.3 and 1.0. It is set equal to whichever of these two limits it exceeds or used as computed if it lies within them. Eqn (22) is then used to estimate the power drawn and eqn (24) is used to provide the residual on the guessed exit enthalpy. The enthalpy residual is incorporated in a Secant method search with the exit enthalpy as the iteration variable. This completes the one-dimensional solution of the quasi-steady-state compressor model. Knowing the power estimated under these conditions, the RLA that the motor would operate at to deliver this power is determined from eqn. (27) and compared with the limiting RLA as described above.

The two-dimensional quasi-steady-state solution of the compressor for when the power is known and the flow-rate is required to be determined, proceeds on the following lines. The two residual required for this solution structure are the enthalpy residual as computed using the 1D solution (Fig. 11) and a power residual computed using eqn. (22). Beginning with guesses for the mass flow rate and exit enthalpy (good starting guesses are the values from the previous time step), a residual is computed for the exit enthalpy. A power residual is computed from eqn. (22) using the guess value of the mass flow rate. Defining these two residuals as the vector function that is desired to be zero, a Newton’s search method is employed (Fig. 12).

The Jacobian of the residuals is calculated and inverted to determine the desired change in the search variables (i.e. the mass flow rate and the exit enthalpy). In order to aid faster convergence, a damping factor is included and the guesses are updated only if the residuals move closer to zero. When the residuals are sufficiently close to zero, the search variables represent the converged values of mass flow rate and exit enthalpy that match the power condition. This scheme has been found to work well in the fault-free condition as well as with

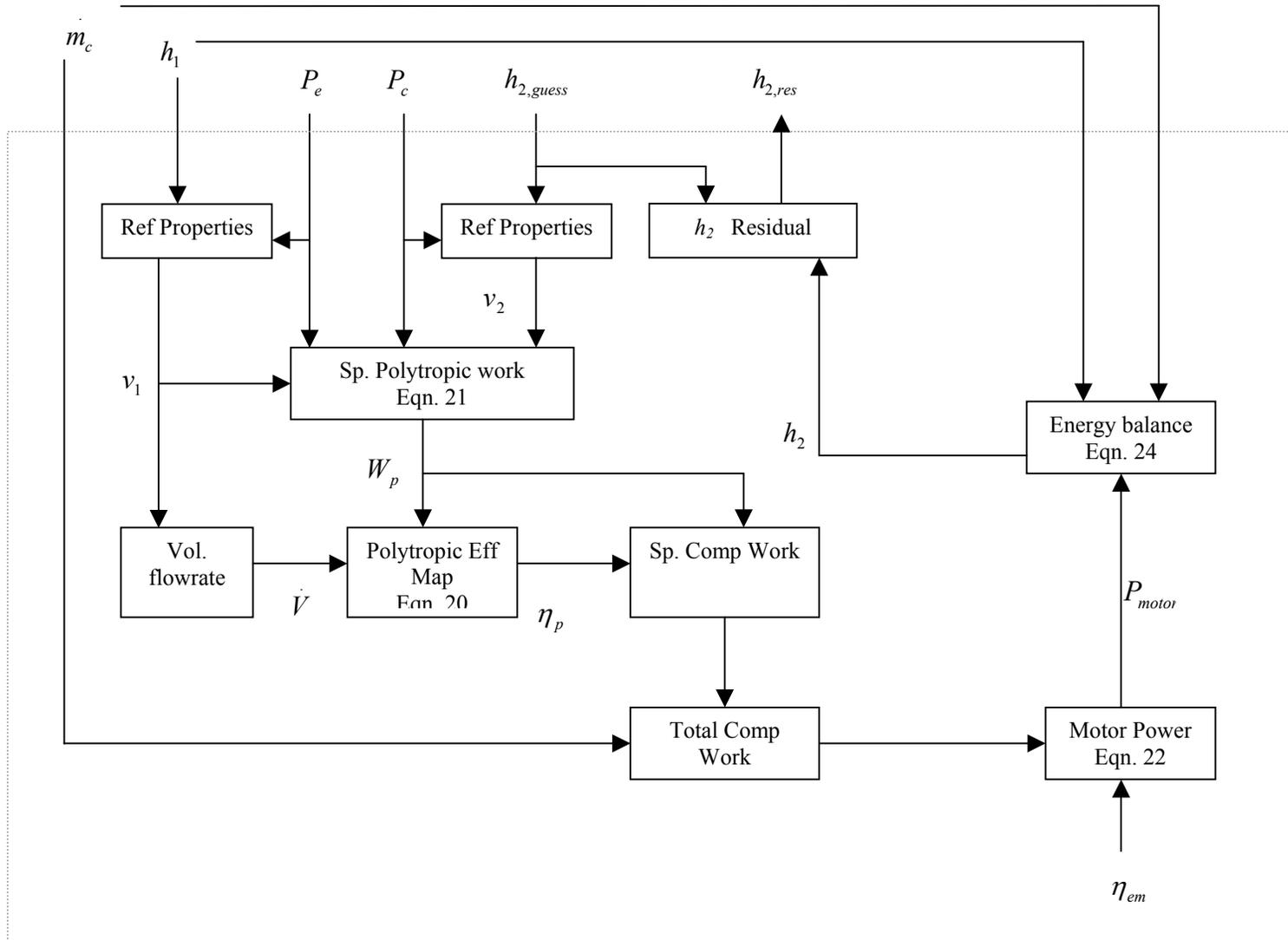


Fig. 11: Compressor exit enthalpy residual computation flow-chart

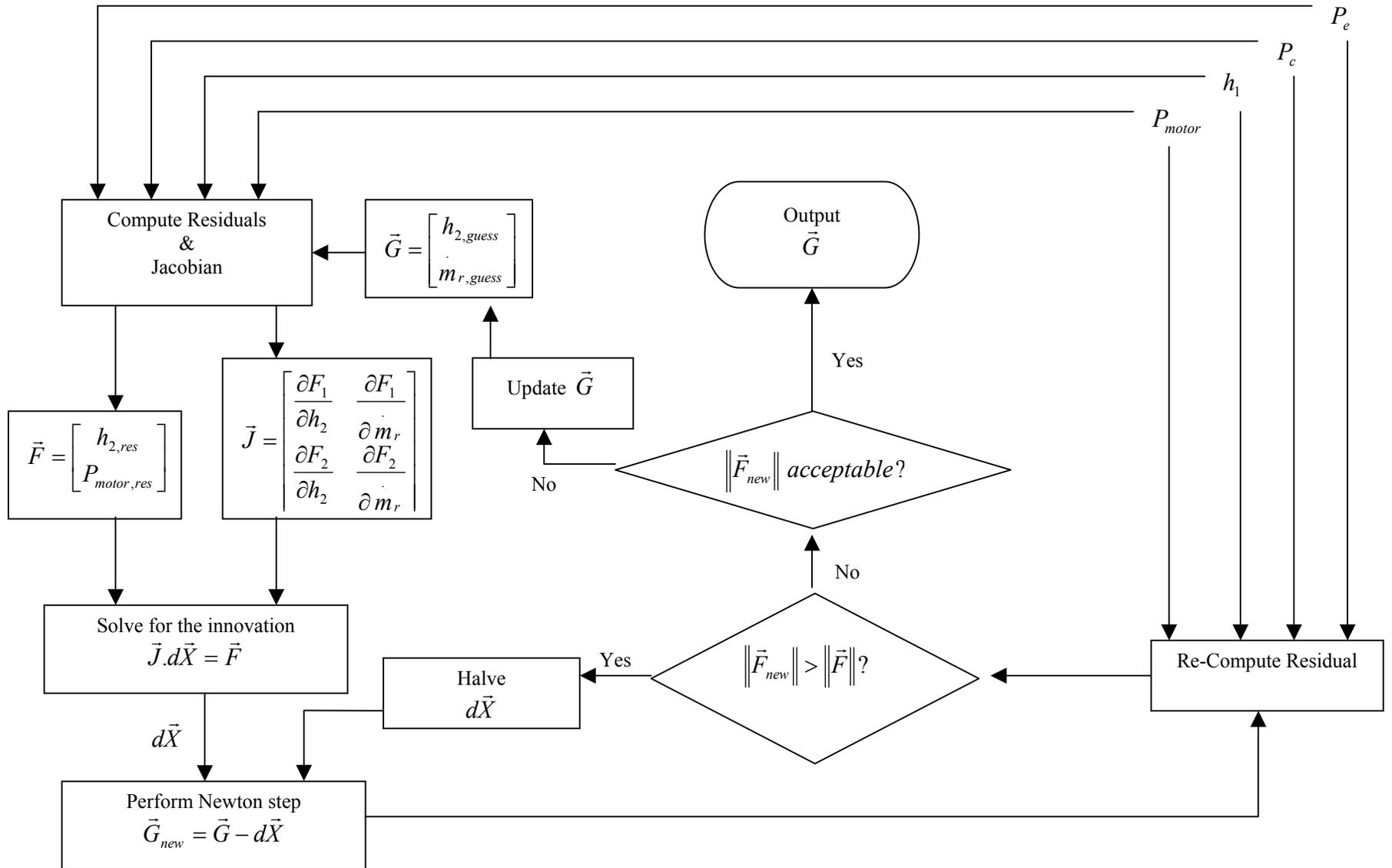


Fig. 12: Solution to the two-dimensional compressor quasi-steady-state model

most of the validated faults. Convergence difficulties were encountered when implementing high (40%) overcharging. Once the compressor equations have converged in one or the other of the above two ways, the losses from the compressor are determined using eqn (23). In summary, the compressor solution begins with inputs of refrigerant inlet state, exit enthalpy and evaporator leaving water temperature and results in outputs of mass flow rate, exit enthalpy, power and losses.

5.4 Valve solution

The valve equations are solved next to determine the mass flow rate through the valve. Using the current evaporator pressure and exit enthalpy, the suction line refrigerant temperature is determined from property routines. This temperature is used in eqn (28), which is integrated forward in time by 1s using a simple, explicit Euler method with a fixed time step of 0.1s. Being a nominally linear, first-order ordinary differential equation, the explicit Euler method is adequate to obtain a stable and accurate solution if the time-step is sufficiently small. The 0.1s integration time step was obtained by continually halving the step until the solution change was within 1e-6. This integration loop executes over the 1s interval of the second-level nested loop described above.

The bulb pressure thus updated, acts against the evaporator pressure and the spring preload and impacts the position of the valve i.e. the displacement, through eqn (29). Eqn. (30) then allows determination of the flow area of the valve. The flow area is constrained to the maximum area (provided during system specification). From current condenser pressure and exit enthalpy, the specific volume of refrigerant in the liquid line is computed, which allows calculation of the mass flow-rate using eqn (31). The flow-rate through the valve is forced to zero for negative pressure differentials.

5.5 Heat exchanger integration

The integration of the heat-exchanger equations, i.e. eqns. (5), (7) and (9), is by a one-step explicit Euler predictor-corrector method which is shown in Figs. 13 & 14. These equations are integrated forward in time by 1s using nominally fixed³ time-steps (0.005s in the

³ Refrigerant inventory error is used to refine the time-step if needed, as explained later.

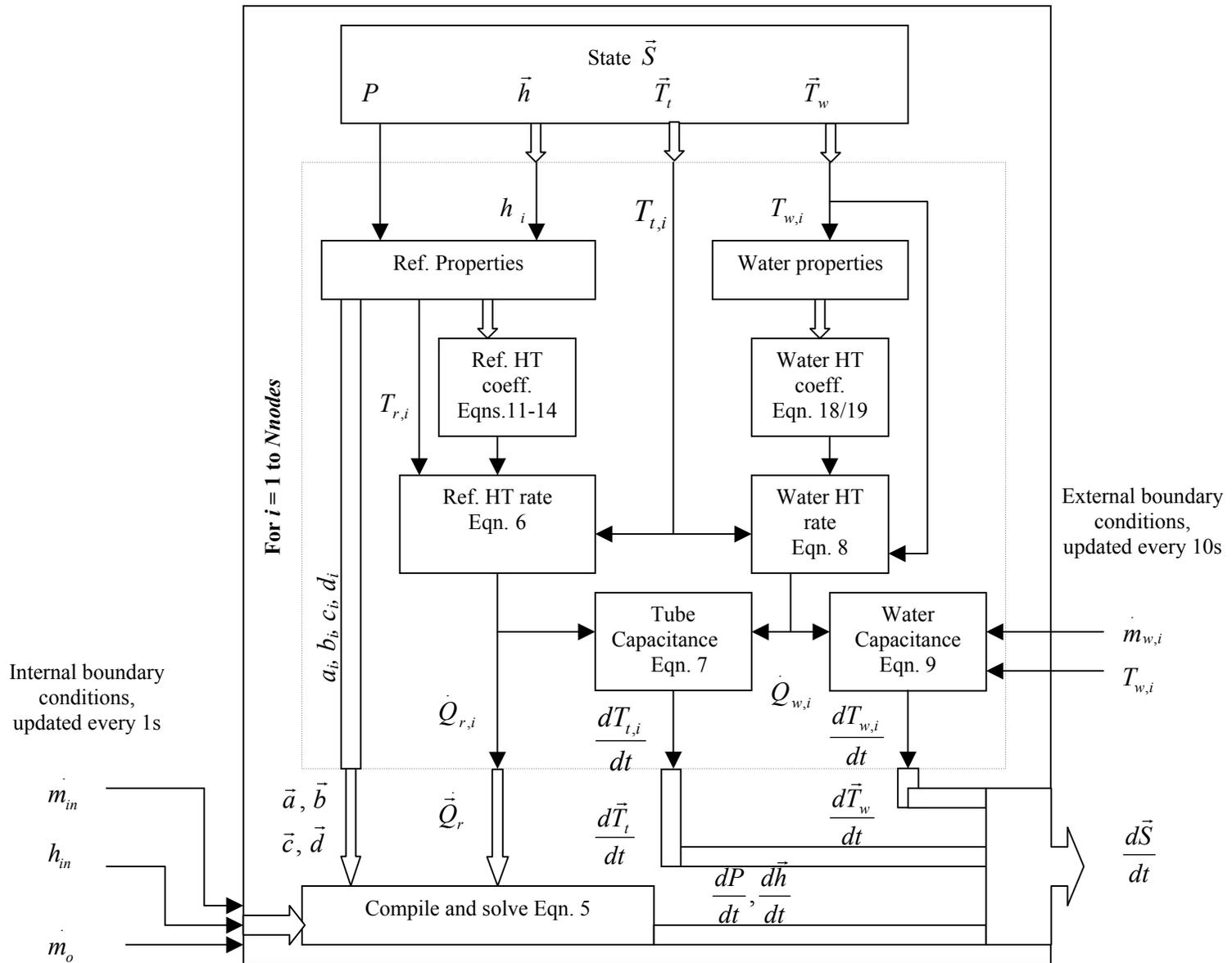


Fig. 13: Information flow within heat-exchanger model

evaporator and 0.025s in the condenser)⁴. As seen in Fig. 13, the state vector \bar{S} for the heat exchanger consists of the pressure, nodal enthalpies, tube temperatures and water temperatures. At any time, the current state is used to determine the refrigerant temperatures and refrigerant-side (eqns. 11-14) and water-side (eqns. 18 & 19) heat transfer coefficients and heat transfer rates (eqns. 6 & 8). This provides sufficient information to construct the coefficient matrix A and input vector B of eqn. (5) and the right-hand-sides of eqns. (7) and (9). These two equations and the solution of eqn. (5) by the LU decomposition algorithm (Hoffman [1992], Press et al[1991]) with successive improvement helps completely determine the heat-exchanger state derivative vector.

The one-step explicit Euler predictor-corrector integration itself proceeds as shown in Fig. 14. From the current heat-exchanger state, the state derivative is computed and integrated forward as the predictor step to the predicted state. At the predicted state, the state derivative is computed again for the corrector step. The corrected state is computed using the average of the two state derivatives. Before this corrected state is accepted, a mass balance check is performed by comparing the difference in the refrigerant quantity in the heat exchanger before and after the integration step with the net refrigerant flow into the heat exchanger. If the absolute difference is beyond a maximum allowable limit, the integration step-size is halved and the corrected state is re-computed. This continues until the mass balance error is below the allowable limit, at which point the corrected state is used to update the heat-exchanger state. This completes one integration step of the heat exchanger and constitutes the innermost loop of the system solution process. The next integration step continues with the same step-size as the previous step (with refinement if any has occurred) and this loop executes until the heat-exchanger state has been integrated forward in time by 1s. Every 1s loop begins with the pre-set time-steps mentioned above.

The one-step correction eliminates the iteration inherent in the otherwise implicit corrector step. This makes for faster execution times, while achieving $O(2)$ global accuracies.

During earlier development of the model, an explicit Euler integration algorithm was investigated. It was found that in order to achieve good accuracy, extremely small time-steps

⁴ The coefficients of eqn. (5) are non-linear as they depend on refrigerant properties and stability analysis is not straightforward. The integration step-sizes used in the heat-exchanger models were obtained by monitoring stability on standalone models of the heat exchangers and continually halving the integration step size.

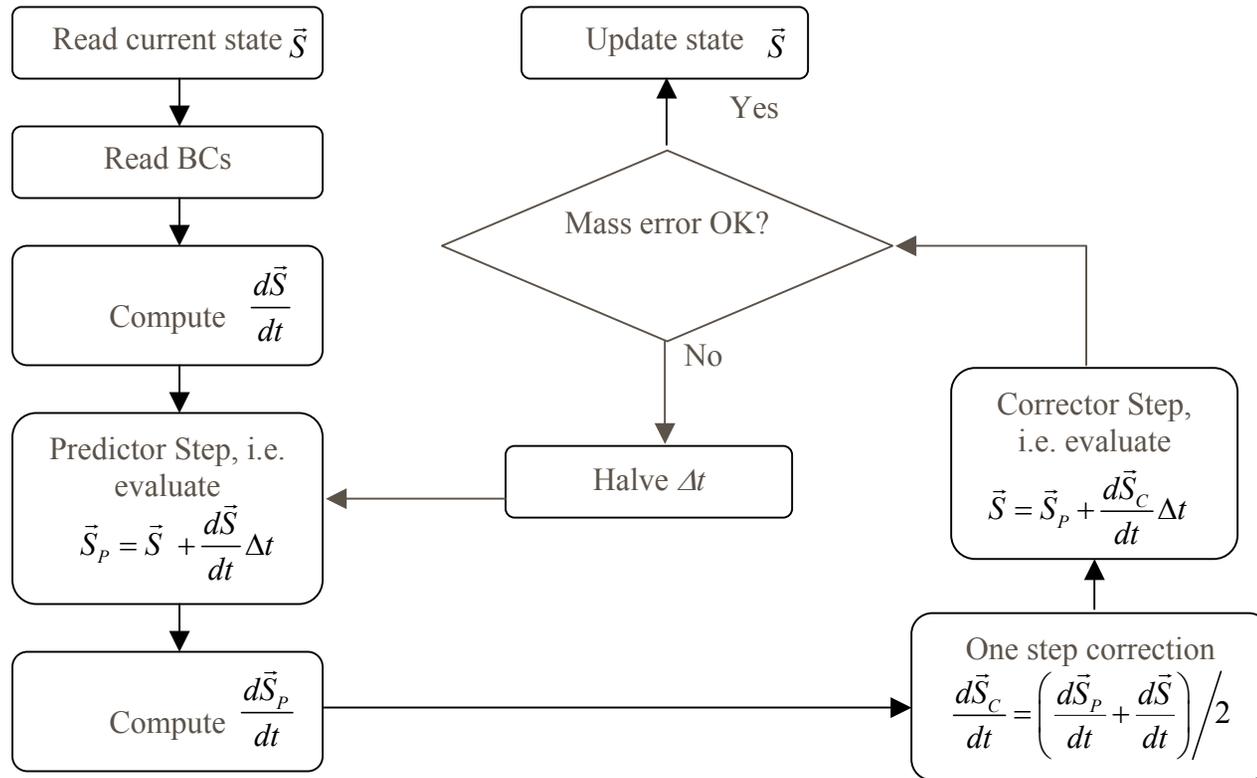


Fig. 14: Heat-exchanger integration algorithm

were required. Consequently, a higher-order algorithm was considered as necessary for the problem. A Runge-Kutta 4th order integration (RK4) was investigated and gave good accuracy but with relatively high computational requirements. The Euler predictor-corrector method also gave good accuracy but with faster computation.

6 Validation

The model was validated using data from the 90 ton McQuay chiller, described in Section 3. The validation of the model was done under the following conditions:

- (a) fault-free;
- (b) 20% reduced condenser and evaporator water flow rates;
- (c) 20% reduced refrigerant charge;
- (d) 20% refrigerant overcharge;
- (e) 45% fouling in condenser.

In each of these cases, the model was run through the complete test cycle described in section 3.3, beginning from start up, through to (but not including) shutdown. Appendix D describes how to use the model software from within a Matlab environment. The plots presented in this section can be re-generated by following the instructions and examples provided in Appendix D.

6.1 Model settings

The heat exchangers were discretized into four nodes each. Early simulations were attempted with two and three nodes in each heat exchanger. It was found that using lesser than four nodes in the condenser resulted in the exit node remaining in a low-quality two-phase condition at steady state, which meant that sub-cooling was zero. Similarly, three nodes in the evaporator were not sufficient to predict the superheat correctly. In order to confirm sufficiency of using four nodes, the model was run using six nodes in each heat exchanger. The increase in accuracy was found to be marginal in comparison with the more significant loss in computation speed. Therefore, the discretization was limited to four nodes. With refrigerant charge related faults introduced however, more nodes were found to be necessary.

The heat transfer coefficients computed using eqns. (12), (13) and (14) required some tuning during validation. This was necessitated by the assumption of smooth tubes in the geometry definition of the heat exchangers. In reality, the tubes in the heat exchangers of the test stand have surface enhancements in the form of rifling on the inside, and corrugations on the outside. In the interest of simplicity, the tuning of the heat transfer coefficients consisted of simply scaling-up the values computed using the aforementioned equations. Table 3 shows the values of the surface enhancement correction factors used in the model.

The condensation correlation correction factor is seen to be the highest. This is possibly due, in part, to the fact that the correlation used applies to laminar flow while the true flow within the condenser is turbulent. Further investigation into alternative correlations for a better condensing heat transfer coefficient is warranted.

Table 3: Surface enhancement correction factors

Heat transfer correlation	Evaporator	Condenser
Refrigerant side		
Sub-cooled	-	$C_{sf,sp} = 1.5$
Two-phase	-	$C_{sf,c} = 22.5$
Superheated	$C_{sf,sp} = 2.5$	$C_{sf,sp} = 6.0$
Water side	$C_{sf,we} = 1.25$	$C_{sf,wc} = 3.0$

6.2 Fault-free condition

Figs. 15, 16 & 17 compare the performance of the model with the measurements with no faults in the system. Fig. 15 shows the performance of the model in predicting the steady-state conditions. Fig. 16 shows the transient performance of the model during start-up and Fig. 17 shows the transient performance during LC9 (please refer to section 3.3 for an explanation of this terminology). This particular transient was selected as it includes changes in all the externally controllable temperatures in the system making it the most dramatic transient. The transient LC18 (a change from SS18 to SS19) is another such variation, but to avoid redundancy this has not been presented here.

It is seen that all steady-state operating conditions are predicted well by the model with the exception of the evaporator pressure, which is significantly under-predicted at low to medium loads and over-predicted at high loads. This is believed to be because of the

approximations that were required in the valve model in the absence of adequate constructional information of the pilot and main valves, or of performance maps. Attempts were made to build maps for the valve from the measurements with little success. Therefore, although the model's predictions of superheat and evaporator pressure are consistent thermodynamically,

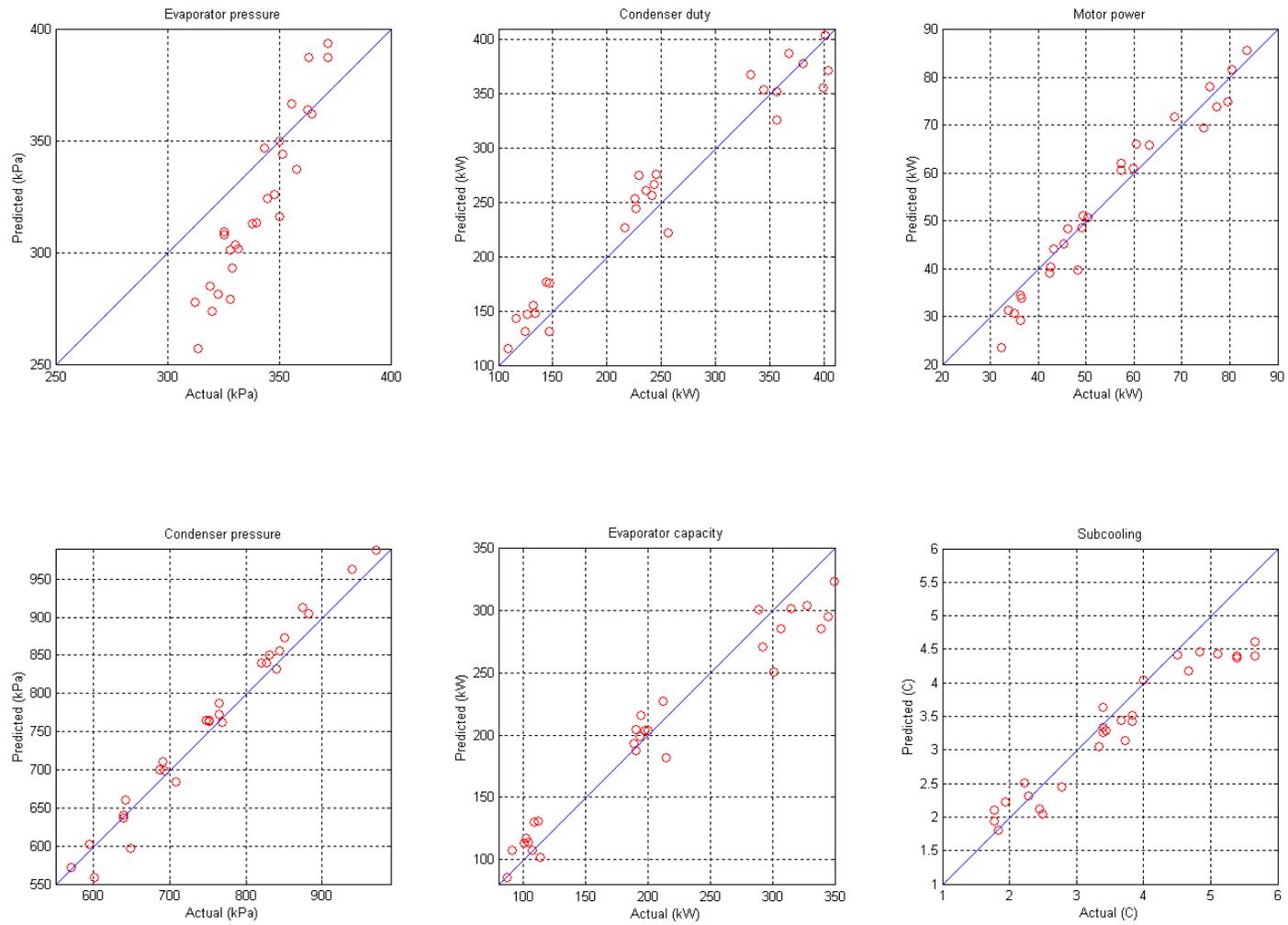


Fig. 15: Steady state model performance in fault-free condition

they do not compare well with the measurements only because the valve model does not truly model the expansion device on the test set-up. From the overall model perspective, however, the model's predictions of superheat and evaporator pressure can be used to observe trends for systems that have thermostatic expansion valves. Further work towards improving the models ability to predict superheat and evaporator pressure is necessary.

Fig. 16 shows the first 2000 seconds of operation after start-up and includes the approach to and achievement of the first steady-state condition. It is seen that all the significant system parameters respond with the correct time-constants and reach the correct steady-state values. The ramp-up of motor power indicates the controller action of constraining the motor current. During the first 100 sec (approximately) after start-up, when the pressure difference across the compressor is low, the polytropic work and volumetric flow rate can be driven to high values. Eqn (20) uses these two parameters to estimate the polytropic efficiency and the conditions noted can result in extrapolation beyond the data ranges used to build the regression. This was found to occur only during the very early start-up. In order to prevent extrapolation, the polytropic efficiency was held constant during this time (see section 5.3). The map was used after the risk of extrapolation was past. The effect of this approach is seen in the compressor power remaining low during the first ~100 sec of the simulation.

Fig. 17 shows the transition of the system LC9, corresponding to the set point temperature changing from 10°C to 7.22°C, the condenser water entering temperature changing from 16.96°C to 29.59°C and the evaporator water entering temperature changing from 11.94°C to 13.42°C.

The absence of the shell thermal mass in the model, which accounts for roughly 18% of the thermal capacitance of the metal and water in the heat-exchanger, appears in the form of relatively faster responses in the pressures, temperatures and motor power. Including the shell dynamics would provide an alternate and parallel path for heat transfer to or from the refrigerant, thereby affecting the pressures change rate. It would also appropriately increase or decrease the amount of heat transferred between the refrigerant and the water thereby affecting the water temperature change rate. The motor power overshoot is possibly a consequence of the absence of any thermal mass in the compressor model. The faster rise in the sub-cooling is a consequence of the faster condenser pressure rise.

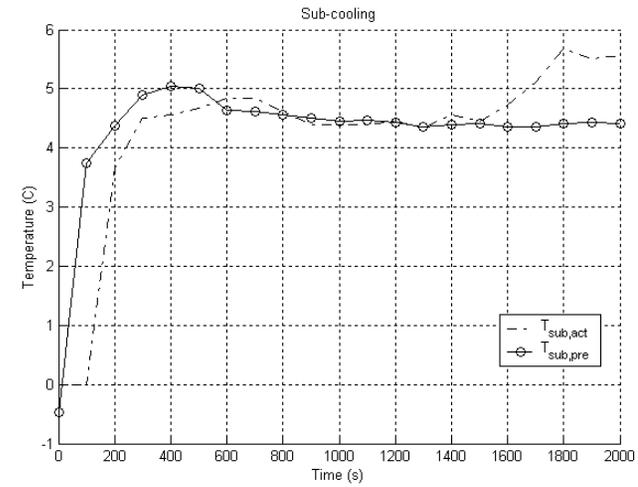
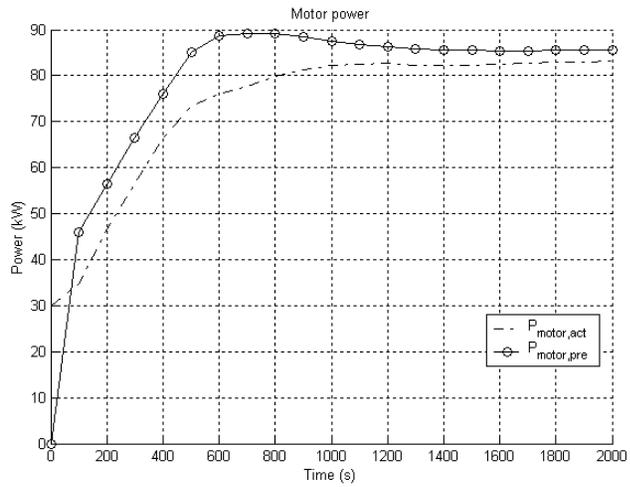
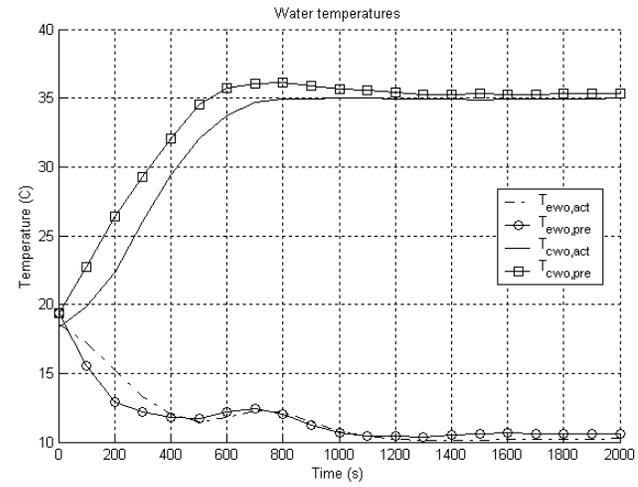
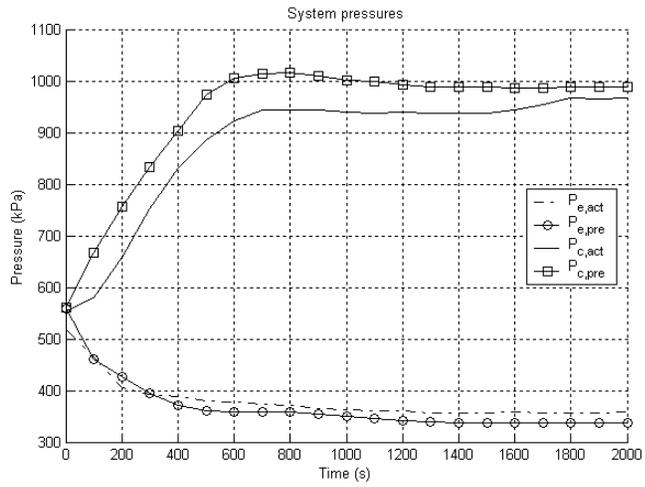


Fig. 16: Start-up model performance in fault-free condition

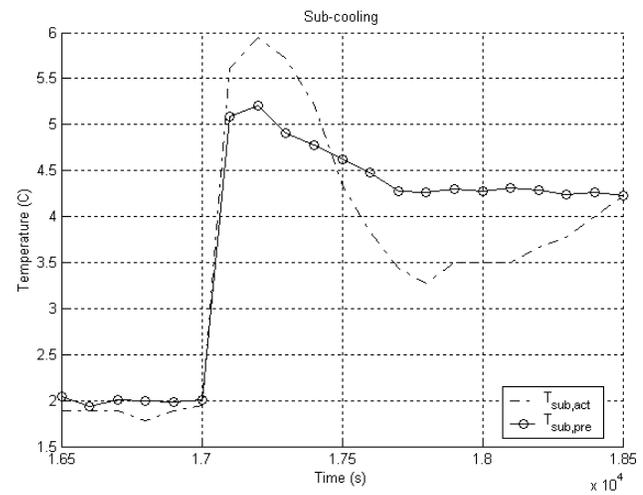
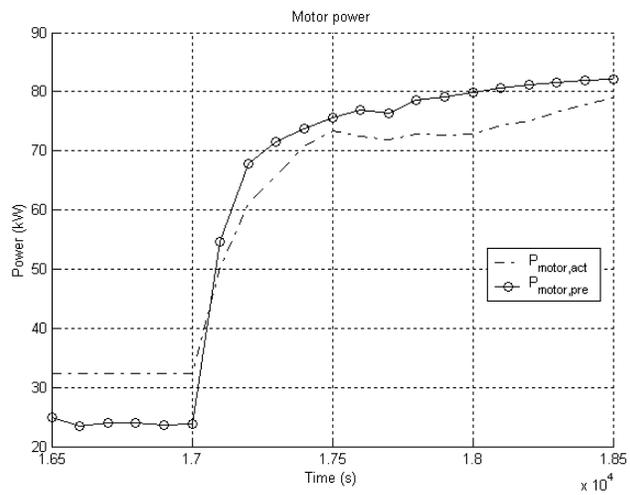
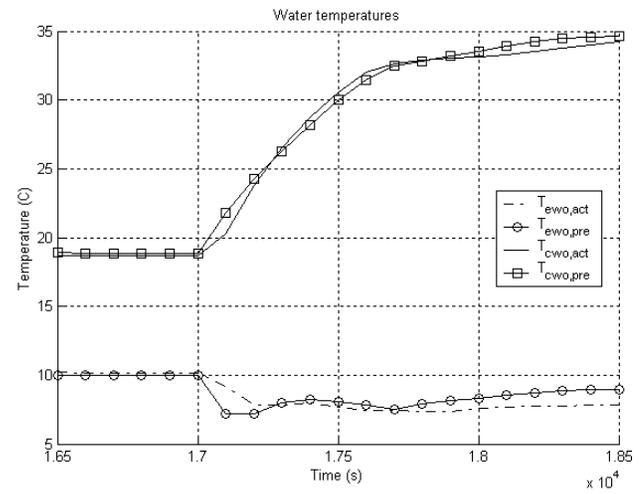
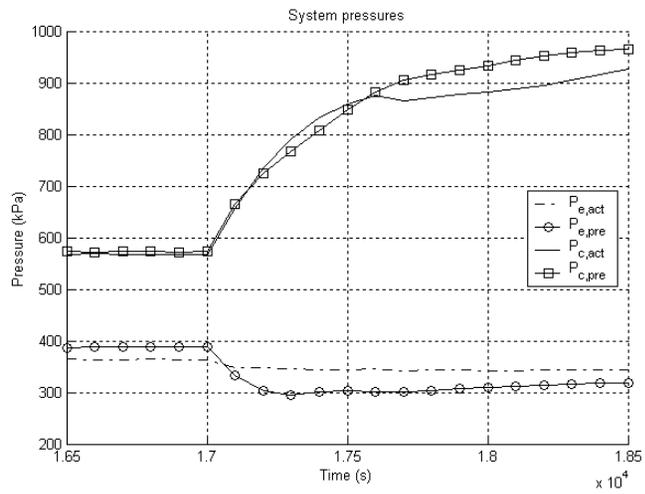


Fig. 17: Load change (LC9) model performance in fault-free condition

The implication of neglecting these thermal capacitances is that the model would tend to marginally over-predict condenser pressures, condenser leaving water temperatures and compressor power during transients. The sub-cooling prediction is impacted as a consequence of the condenser pressure. The evaporator pressure also would respond quicker, resulting in possible under-prediction during transients. However, the steady-state behavior would in no way be affected by this assumption.

Fig. 18 shows the numerical conservation of refrigerant charge during a complete test cycle that spans 14hrs and 14 minutes (real-time). The first plot shows the early migration of the refrigerant from the evaporator to the condenser. This is a transient that lasts about 90s. The second plot shows the subsequent, sustained distribution of refrigerant between the two heat-exchangers. The small fluctuations correspond to the load changes. The total system charge is seen to be perfectly constant. This is a result of the iterative step-sizing used in the heat-exchanger integration (Fig. 14).

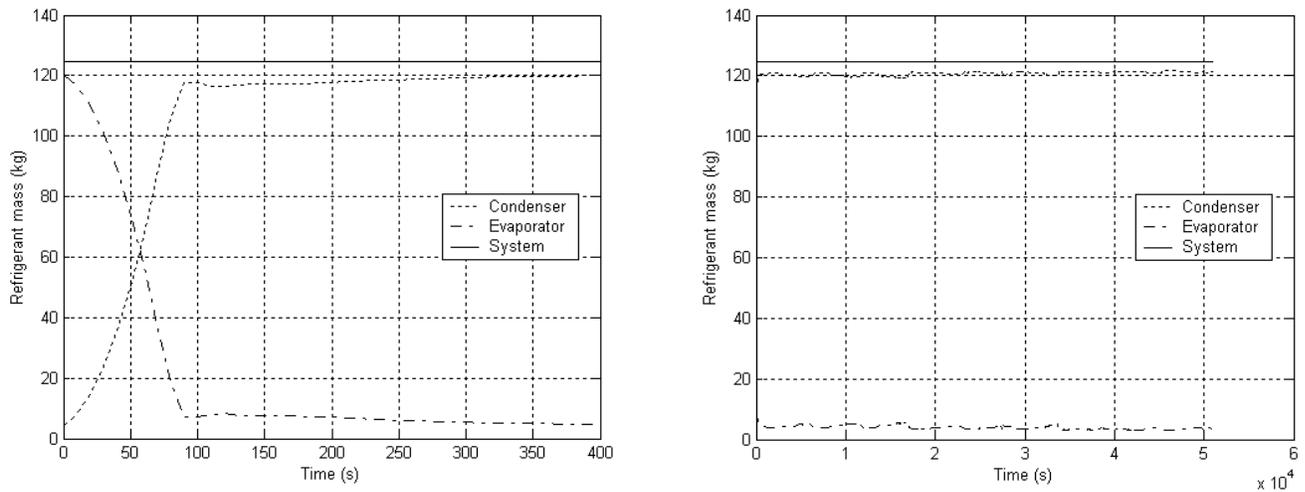


Fig. 18: Refrigerant charge during system simulation

6.3 Reduced water flow rates

In order to validate the model for reduced water flow rates through the heat exchangers, the data available from the test condition of combined reduction of evaporator and condenser water flow rates (by 20%) was used. The model was run through the complete test cycle and Figs. 19, 20 & 21 show respectively, the steady-state, start-up and LC9 load-change performances of the model with this fault. As with the fault-free condition, it is seen that the steady-state evaporator pressure is similarly under-predicted at low loads, while the other parameters are well captured at steady-state. The start-up and load change transients are also seen to be reproduced reasonably accurately. Table 4 below compares the measured change in system performance with this fault from the fault-free condition measurements with the predicted deviations of system performance with this fault, from the fault-free condition predictions, at the 27 steady-state conditions.

Table 4: Deviation of steady-states from fault-free with reduced flows

SS No.	Evap. Pr (kPa)		Cond. Pr (kPa)		Cond. Duty (kW)		Evap. Cap. (kW)		Motor Power (kW)		Sub-cooling (C)	
	Act	Pre	Act	Pre	Act	Pre	Act	Pre	Act	Pre	Act	Pre
1	-2.1	0.5	10.3	49.0	-25.9	-8.5	-33.5	-12.4	-4.9	2.3	0.2	0.8
2	-2.1	8.1	3.4	2.1	-28.7	-48.7	-29.4	-41.4	-3.2	-6.8	0.5	0.3
3	-4.8	4.7	-6.9	-4.1	-18.7	-35.7	-16.3	-31.5	-3.8	-5.6	0.1	0.1
4	0.7	-6.4	6.9	32.3	-44.3	-17.2	-49.5	-17.0	-6.8	1.0	0.0	0.7
5	-1.4	3.4	6.9	16.8	-24.2	-23.6	-23.1	-21.6	-1.9	-1.8	0.4	0.4
6	-2.8	1.9	6.9	1.9	-9.3	-19.6	-13.6	-15.4	-1.0	-5.6	0.3	0.0
7	0.0	17.0	6.9	17.4	-35.0	-45.7	-38.8	-38.0	-4.3	-6.6	0.5	0.4
8	-1.4	4.6	9.7	13.7	-22.4	-24.8	-26.0	-23.2	-1.6	-1.3	0.3	0.4
9	-2.1	3.0	3.4	5.9	-10.8	-23.0	-14.1	-19.2	-0.5	-4.0	0.1	0.1
10	-2.1	-6.1	-3.4	34.8	-27.6	-18.1	-31.8	-17.0	-3.4	0.8	-0.1	0.7
11	-3.4	7.7	-3.4	7.1	-32.2	-37.5	-32.8	-30.7	-3.9	-5.5	0.4	0.3
12	-3.4	7.6	0.0	-9.0	-21.7	-45.0	-22.2	-37.5	-3.1	-7.4	0.1	0.0
13	2.1	-1.3	6.9	38.1	-51.0	-11.8	-49.7	-12.1	-8.0	0.2	0.4	0.8
14	-2.1	8.5	3.4	15.3	-24.3	-17.4	-24.6	-15.7	-2.1	-1.9	0.3	0.5
15	-3.4	3.8	10.3	-1.3	-14.3	-28.0	-15.9	-23.5	-1.4	-3.8	0.2	0.1

Table 4 (Contd.): Deviation of steady-states from fault-free with reduced flows

SS No.	Evap. Pr (kPa)		Cond. Pr (kPa)		Cond. Duty (kW)		Evap. Cap. (kW)		Motor Power (kW)		Sub-cooling (C)	
	Act	Pre	Act	Pre	Act	Pre	Act	Pre	Act	Pre	Act	Pre
16	1.4	-6.4	13.8	31.3	-29.3	-20.2	-33.7	-18.7	-2.8	-0.1	0.6	0.7
17	0.7	14.4	6.9	2.1	-26.2	-39.2	-27.2	-32.7	-2.0	-5.1	0.6	0.3
18	1.4	4.2	0.0	-0.2	-13.7	-24.8	-15.9	-24.2	-1.3	-2.6	0.0	0.0
19	-1.4	-8.9	6.9	12.2	-54.9	-14.7	-52.0	-13.6	-10.7	-0.1	1.2	0.7
20	-3.4	8.4	-3.4	12.4	-41.2	-26.1	-38.2	-19.6	-4.6	-3.9	0.1	0.4
21	-2.1	8.5	0.0	-3.2	-19.1	-33.3	-17.8	-27.3	-2.8	-5.2	0.4	0.1
22	-0.7	-7.2	13.8	24.6	-35.9	-17.6	-33.2	-18.9	-5.0	0.5	0.4	0.7
23	-5.5	4.6	0.0	0.4	-32.1	-37.3	-34.3	-30.9	-3.3	-5.0	0.3	0.3
24	0.0	6.4	0.0	11.1	-17.4	-21.7	-17.8	-20.1	-1.9	-2.3	0.3	0.1
25	-2.1	-6.2	-41.4	22.8	-67.2	-16.0	-61.7	-15.8	-10.6	0.4	-0.4	0.7
26	-6.9	4.1	-47.6	10.2	-67.5	-24.7	-59.1	-22.5	-8.7	-2.0	-0.2	0.4
27	0.0	7.4	-37.9	10.0	-29.8	-19.9	-29.6	-18.1	-4.6	-1.6	-1.0	0.1

When the water flow-rate in a heat exchanger is reduced, the heat transfer coefficient on the water side drops because of the reduction in flow velocity. This results in a reduced rate of heat transfer to the water for the same temperature differential. Therefore, in order to achieve the desired rate of heat transfer, the temperature difference between the refrigerant and the water needs to rise and the condenser temperature (and therefore pressure) rises. For most of the conditions, the condenser pressure change is seen to be a positive one, indicating that it has risen. The few negative changes are fairly small in magnitude and are likely due to other effects.

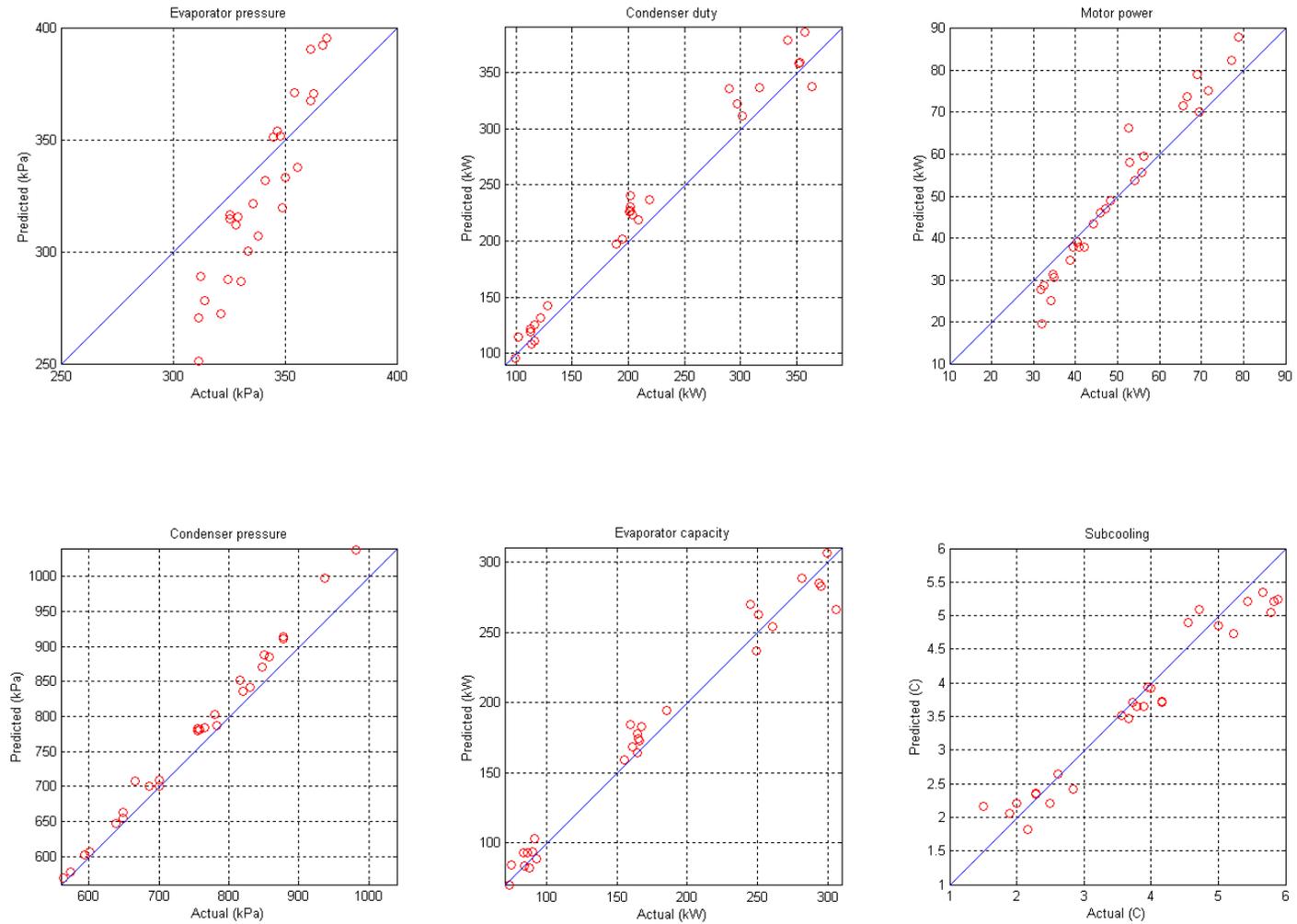


Fig. 19: Steady-state model performance with 20% reduction in water flow-rates

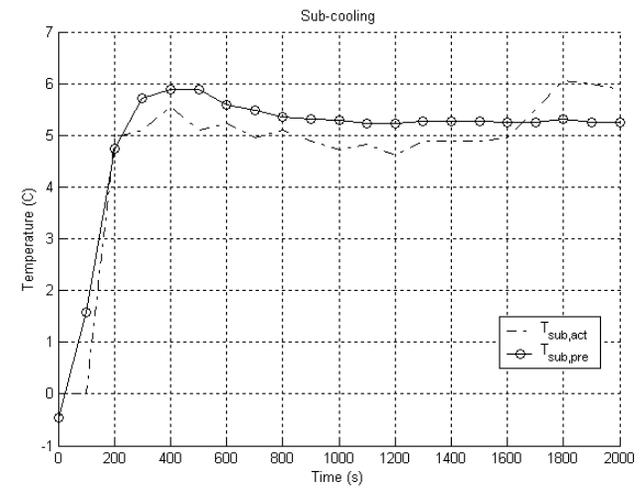
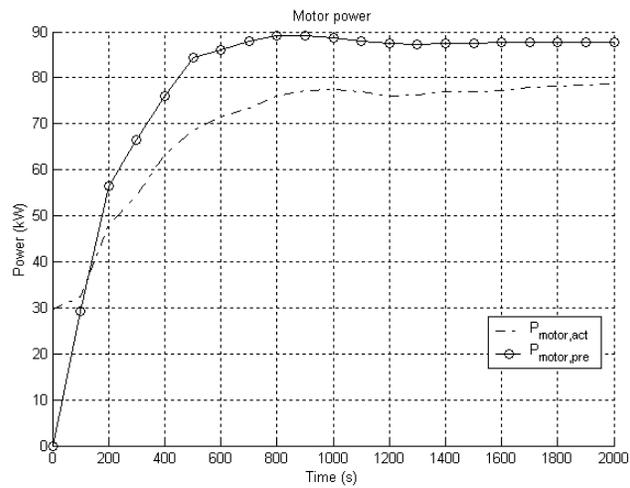
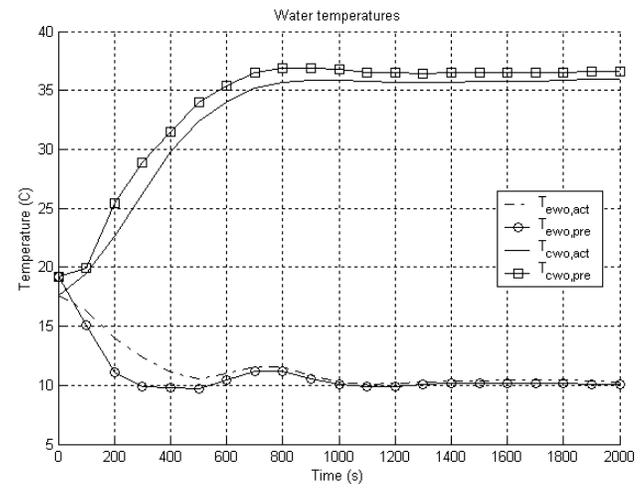
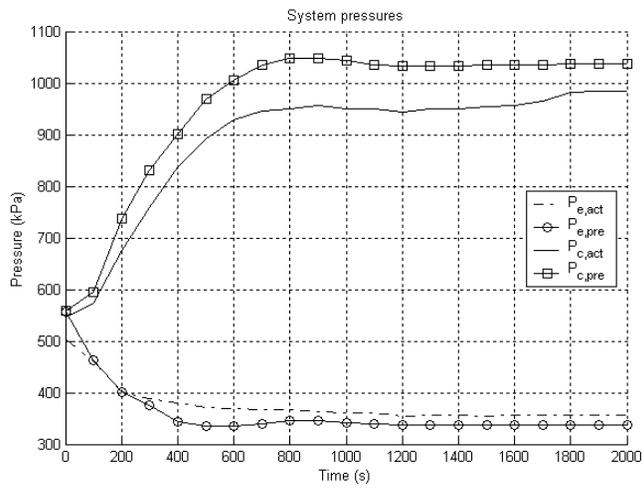


Fig. 20: Start-up model performance with 20% reduction in water flow-rates

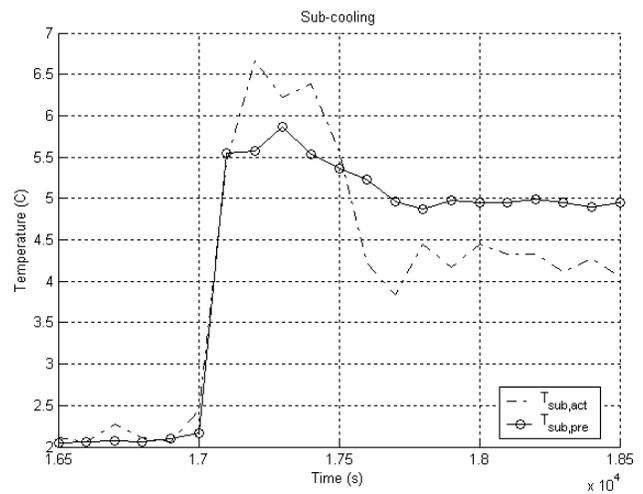
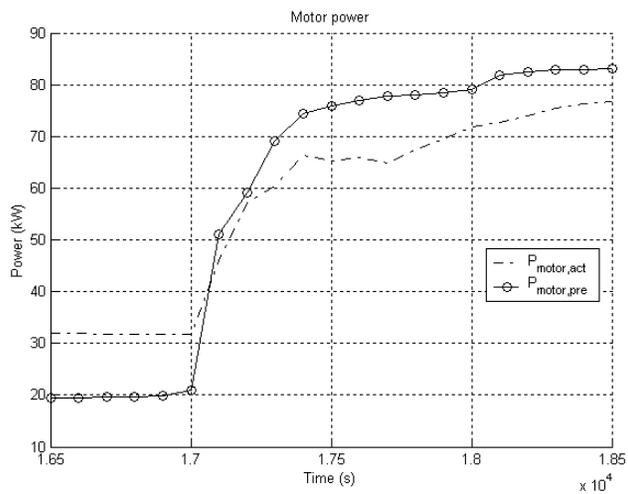
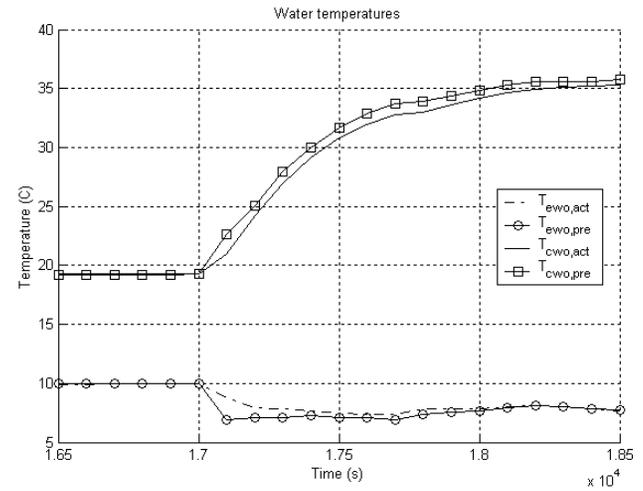
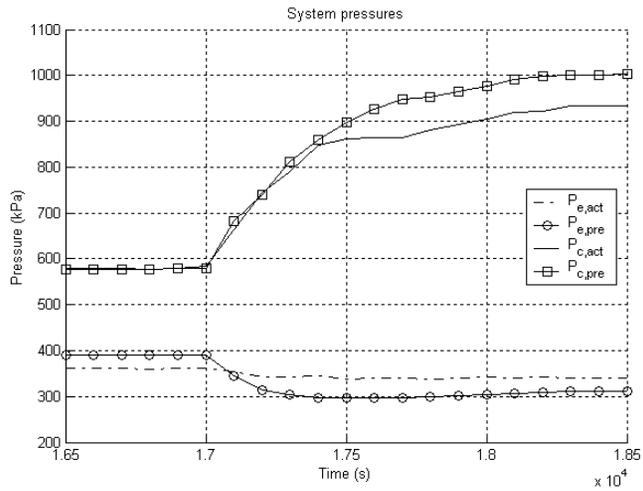


Fig. 21: Load change (LC9) model performance with 20% reduction in water flow-rates

6.4 Refrigerant under-charge

As described within section 4.7, the refrigerant charge is enforced on the system when the pressures and enthalpy distribution in both the heat exchangers are specified. Therefore, to initialize the system to a lower refrigerant charge, the initial enthalpy distribution and pressure are determined corresponding to a lower charge of 99.76 kg (80% of 124.7 kg). From the severity levels for which data was available, the most severe condition corresponding to a 40% refrigerant loss was initially chosen and the model was executed through the complete test cycle. However, it was found that the model predictions with 4 nodes in each heat-exchanger, were significantly different from the measurements. Therefore, the severity level in validation was reduced to a 20% loss of refrigerant. Figs. 22, 23 & 24 present the start up and transient performance associated with LC9 respectively with the refrigerant charge reduced by 20%.

Table 5 compares the change in system performance due to the charge reduction, as predicted by the model, with the measurements. As expected, it is seen that the condenser pressure shows an almost uniform drop through all steady-state operating conditions. The reduced charge also suppresses the sub-cooling. From Fig. 22, it is seen that the major system parameters (with the exception of evaporator pressure) are well predicted. The sub-cooling however shows an under-prediction. This indicates that the model is more sensitive to the refrigerant charge than the actual system and although the change in sub-cooling is in the expected direction, its magnitude is more than measured.

Table 5: Deviation of steady-states from fault-free with reduced charge

SS No.	Evap. Pr (kPa)		Cond. Pr (kPa)		Cond. Duty (kW)		Evap. Cap. (kW)		Motor Power (kW)		Sub-cooling (C)	
	Act	Pre	Act	Pre	Act	Pre	Act	Pre	Act	Pre	Act	Pre
1	-1.4	0.0	6.9	-26.7	12.7	-0.4	10.1	0.2	1.7	-1.2	0.1	-2.3
2	-4.8	-1.5	10.3	-16.6	7.5	2.8	9.8	1.3	1.5	-0.4	0.1	-1.7
3	-2.8	1.3	3.4	-3.6	1.6	-5.9	4.9	-5.8	0.7	-0.9	-0.1	-0.9
4	-2.8	-0.8	6.9	-21.1	8.4	-1.7	5.8	0.5	1.4	-0.9	0.6	-2.3
5	-8.3	-2.1	10.3	4.5	17.6	7.1	15.7	5.6	2.6	1.4	0.3	-1.5
6	-11.7	-2.4	6.9	10.0	1.8	14.4	4.3	10.8	0.6	2.3	0.2	-0.6
7	-5.5	-6.4	6.9	5.6	14.2	27.5	13.0	21.4	2.0	6.7	0.2	-2.0
8	-12.4	-4.4	6.2	3.7	14.1	14.4	13.3	11.6	2.0	1.5	0.2	-1.3

Table 5 (Contd.): Deviation of steady-states from fault-free with reduced charge

SS No.	Evap. Pr (kPa)		Cond. Pr (kPa)		Cond. Duty (kW)		Evap. Cap. (kW)		Motor Power (kW)		Sub-cooling (C)	
	Act	Pre	Act	Pre	Act	Pre	Act	Pre	Act	Pre	Act	Pre
9	-13.1	-2.8	10.3	16.4	9.8	22.0	7.3	15.8	1.5	6.6	0.3	-0.5
10	-2.1	0.6	0.0	-29.1	0.6	1.6	-0.8	4.3	-0.9	-1.1	0.7	-2.1
11	-4.8	-15.4	6.9	7.9	5.2	29.7	6.5	23.2	1.5	7.4	0.2	-1.5
12	-6.2	-3.8	6.9	7.4	-2.6	19.2	0.6	15.4	0.1	3.3	-0.6	-1.0
13	-3.4	6.1	13.8	-11.5	16.2	7.0	11.0	5.8	4.1	0.0	0.8	-2.2
14	-4.1	-8.9	6.9	8.0	8.9	27.8	9.4	22.4	1.8	5.1	-0.1	-1.4
15	-8.3	1.0	13.8	-0.2	1.6	-5.0	5.5	-5.0	1.1	-0.5	0.1	-0.9
16	-3.4	12.3	44.1	11.1	28.8	8.8	27.5	9.3	9.4	1.6	1.4	-2.2
17	-1.4	0.0	6.9	-26.7	12.7	-0.4	10.1	0.2	1.7	-1.2	0.1	-2.3
18	-4.8	-4.6	17.2	7.1	25.3	19.5	19.8	16.6	3.7	3.3	0.3	-1.6
19	-4.1	0.6	13.8	8.3	15.4	-1.9	11.1	-3.9	1.8	0.0	-0.4	-1.0
20	-4.8	7.7	24.1	-9.3	-5.2	10.0	-6.6	10.2	0.7	0.2	0.8	-2.0
21	-6.9	3.7	-13.1	-11.0	-6.1	8.9	-4.2	9.6	-1.2	0.5	-0.7	-1.6
22	-3.4	4.4	-13.8	-16.7	2.2	-14.9	-0.1	-11.5	-0.4	-2.8	-1.4	-1.2
23	-6.9	0.9	-3.4	-22.1	0.7	2.1	3.0	3.0	0.9	-1.0	0.0	-2.1
24	-7.6	-3.4	-10.3	-0.7	5.2	13.9	5.4	12.1	0.2	2.5	-1.8	-1.5
25	-2.1	0.5	-0.7	15.1	3.9	1.1	2.3	-0.3	0.4	0.8	-0.7	-0.9
26	-8.3	2.7	-41.4	-7.7	-8.7	-0.3	-7.1	0.6	-3.3	-0.5	-1.1	-2.2
27	-13.8	-17.0	-3.4	60.6	-4.7	77.7	-1.1	62.0	-0.5	14.7	-0.2	-1.1

The transient responses as shown in Figs. 23 & 24 show good agreement with the measurements.

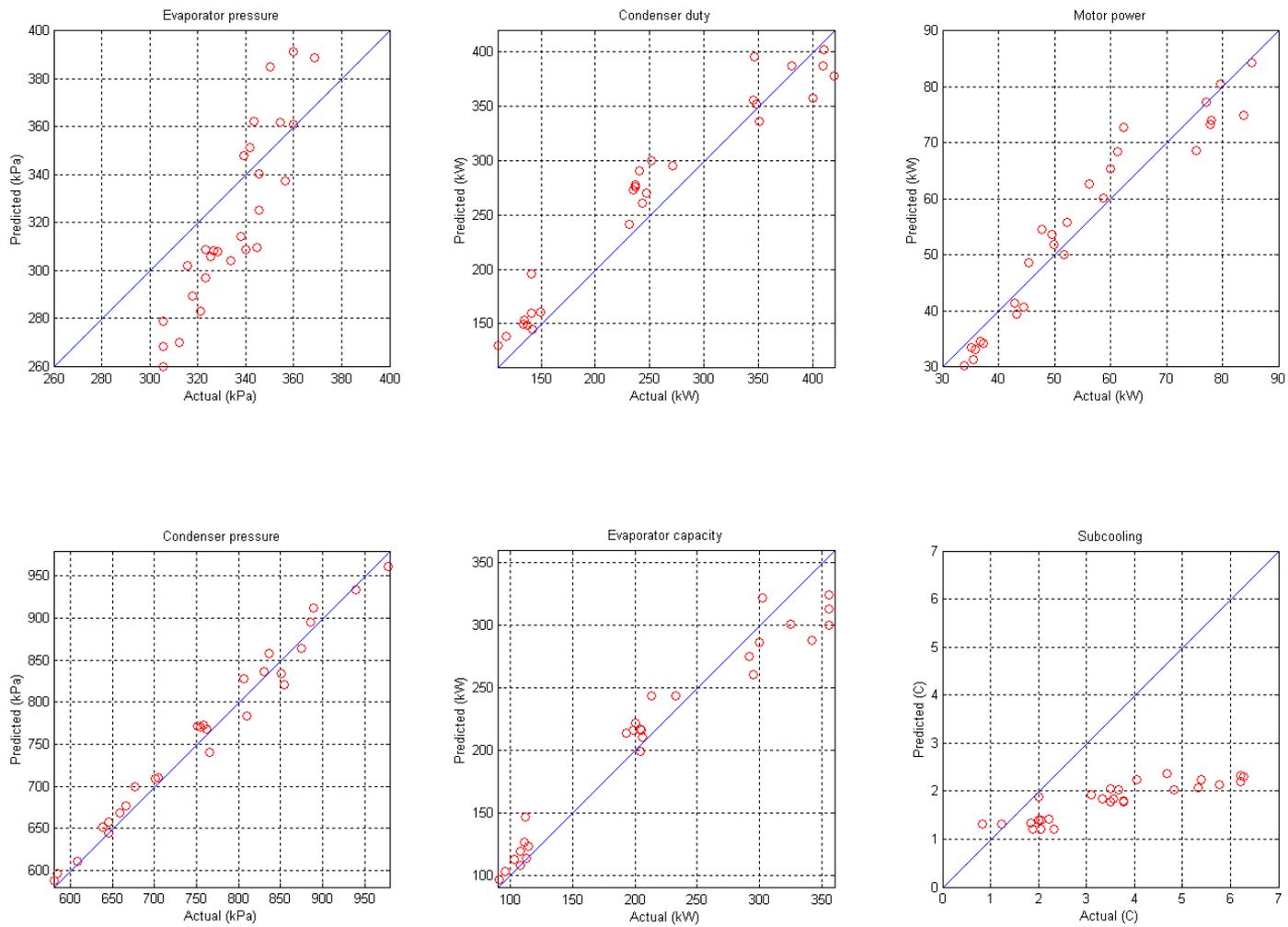


Fig. 22: Steady-state model performance with 20% reduction in charge

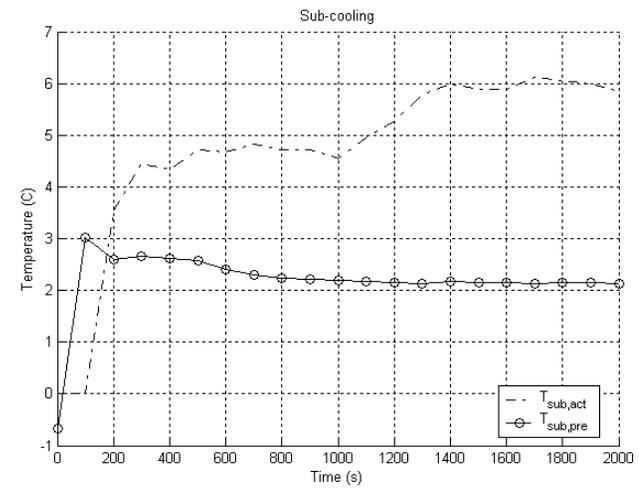
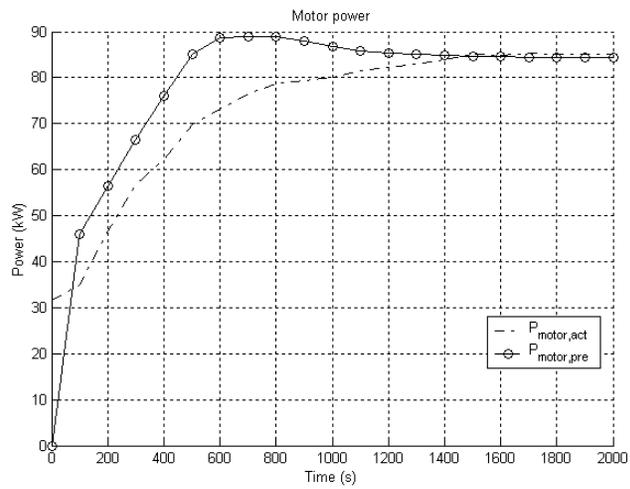
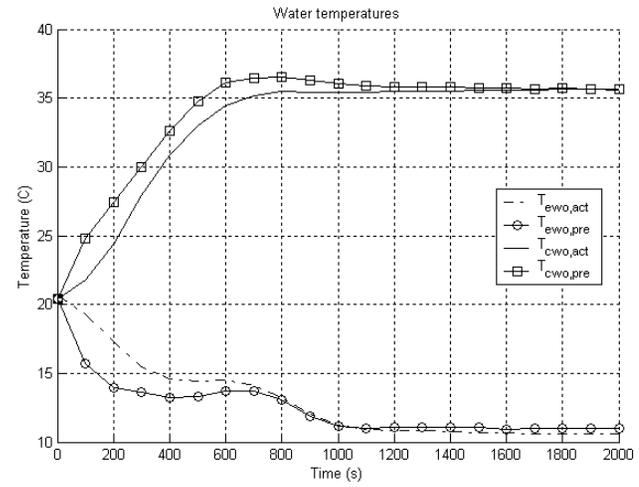
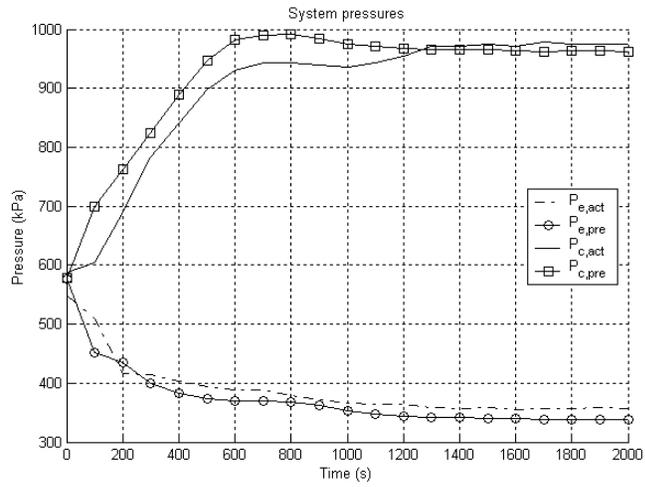


Fig. 23: Start-up model performance with 20% reduction in refrigerant charge

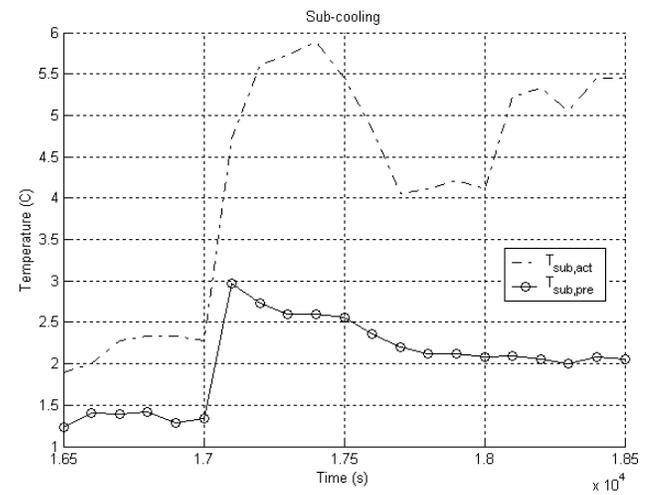
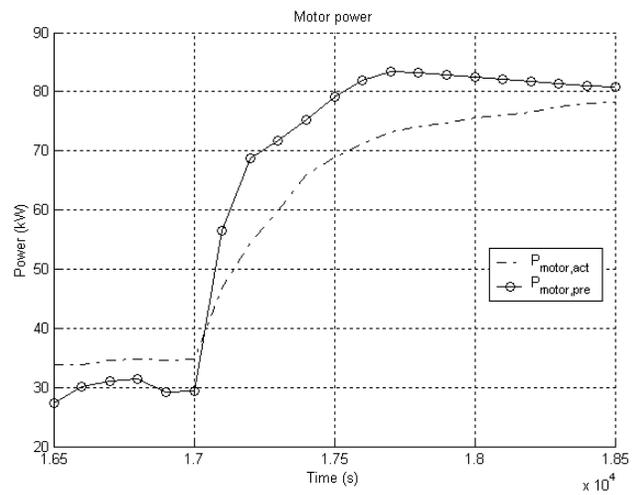
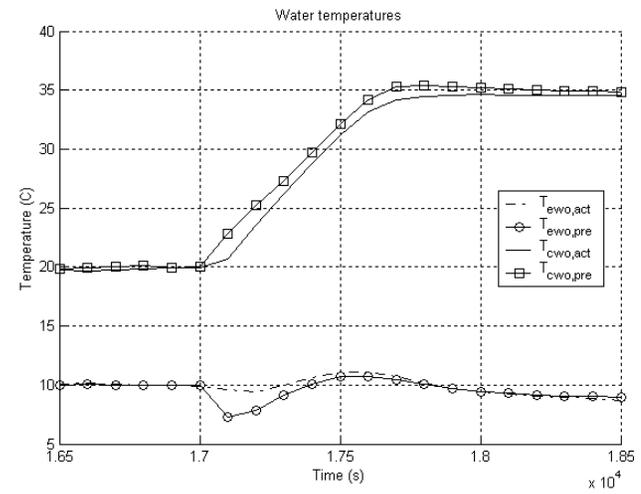
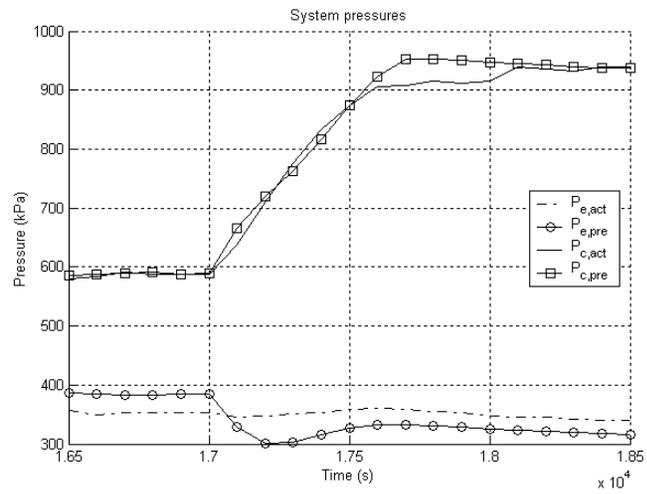


Fig. 24: Load change (LC9) model performance with 20% reduction in refrigerant charge

6.5 Refrigerant overcharge

As with the under-charging fault, some validation attempts were made to capture 40% refrigerant overcharge. However, in addition to the discretization problem encountered above when simulating 40% undercharge, additional convergence issues in the compressor's two-dimensional quasi-steady-state map were encountered. The simulation was repeated by reducing the fault severity to 20% (i.e. 149.64 kg) refrigerant overcharge. Figs. 25, 26 & 27 show the performance of the model under this condition. Table 6 compares the change in system performance between the model and the measurements at the 27 steady-state conditions in the test cycle.

Table 6: Deviation of steady-states from fault-free with excess charge

SS No.	Evap. Pr (kPa)		Cond. Pr (kPa)		Cond. Duty (kW)		Evap. Cap. (kW)		Motor Power (kW)		Sub-cooling (C)	
	Act	Pre	Act	Pre	Act	Pre	Act	Pre	Act	Pre	Act	Pre
1	10.3	20.9	61.4	223.2	-19.3	-6.7	-44.7	-17.4	-5.4	9.3	3.8	9.0
2	-7.6	-3.5	108.9	171.5	40.2	6.3	13.0	-6.3	9.2	10.0	4.3	7.1
3	2.1	0.2	62.1	96.8	19.1	3.2	1.3	-0.4	3.2	3.6	2.7	4.3
4	-1.4	18.7	113.1	226.0	11.3	3.1	-27.8	-8.0	5.1	12.7	5.6	9.5
5	-6.9	2.9	88.9	158.1	33.0	9.5	7.3	0.8	5.5	8.8	4.3	6.7
6	-6.2	0.5	47.6	66.3	16.4	4.3	4.3	-0.1	1.9	1.6	2.3	3.1
7	-5.5	13.1	102.7	236.9	45.3	28.8	0.8	10.2	5.7	19.0	5.7	9.5
8	-9.0	1.5	71.7	132.3	35.4	17.7	6.7	11.2	4.3	8.0	4.1	6.4
9	0.0	-0.2	44.8	86.1	17.7	15.8	1.7	6.0	1.9	9.5	2.6	4.1
10	9.0	17.0	54.5	215.8	-42.8	-0.2	-61.3	-9.7	-7.6	11.6	4.0	8.6
11	-4.8	-8.1	88.9	205.5	23.9	27.3	-4.4	11.4	6.0	16.4	4.1	7.6
12	-4.8	2.8	51.7	97.5	18.0	-7.3	0.1	-10.5	2.7	2.5	2.3	4.4
13	6.2	24.5	95.8	224.5	-3.2	5.8	-40.4	-5.5	-0.1	12.5	5.7	9.0
14	-0.7	-8.4	95.8	254.8	49.3	99.5	19.3	68.9	7.7	30.1	4.4	9.2
15	-5.5	2.1	54.5	75.4	14.4	0.1	2.9	-1.5	1.9	2.4	2.3	3.9
16	-2.1	4.4	109.6	146.7	21.1	-15.7	-18.5	-19.6	6.0	5.1	6.2	8.5
17	4.1	16.3	88.9	118.6	26.1	-16.8	-2.6	-19.5	4.4	3.4	4.2	6.2
18	-0.7	4.2	61.4	60.8	29.8	-12.2	10.2	-14.4	3.3	0.2	2.6	3.3
19	9.7	15.9	99.3	176.3	-33.9	-4.3	-56.6	-11.3	-4.2	7.5	5.1	7.9
20	-9.0	7.4	65.5	182.7	21.9	20.2	4.4	8.1	5.2	14.0	2.4	7.5

Table 6 (Contd.): Deviation of steady-states from fault-free with excess charge

SS No.	Evap. Pr (kPa)		Cond. Pr (kPa)		Cond. Duty (kW)		Evap. Cap. (kW)		Motor Power (kW)		Sub-cooling (C)	
	Act	Pre	Act	Pre	Act	Pre	Act	Pre	Act	Pre	Act	Pre
21	-2.1	5.8	27.6	92.4	5.1	-4.0	3.2	-6.1	1.5	3.1	1.1	4.6
22	9.0	13.7	102.7	169.0	-3.5	-8.7	-32.0	-16.1	0.2	7.1	5.5	8.3
23	-0.7	0.6	71.7	145.0	22.4	17.2	2.6	5.1	4.2	10.9	2.3	6.9
24	3.4	5.1	37.2	82.2	5.0	-4.7	1.5	-6.3	1.7	2.2	0.9	3.8
25	-1.4	14.1	85.5	159.8	9.1	-13.5	-24.8	-18.9	1.5	6.3	5.2	8.1
26	-1.4	-9.6	75.8	202.6	29.5	66.5	0.0	45.6	4.1	20.8	3.7	7.8
27	-5.5	-4.0	57.9	128.8	22.4	30.4	5.5	22.8	2.6	7.8	2.2	4.7

The model is seen to severely over-predict the condenser pressure, and therefore the sub-cooling, with this system charge, while still achieving the appropriate capacities and power consumption, especially at lower loads. This is indicative of a discretization issue and prompts the use of a finer grid at least in the condenser.

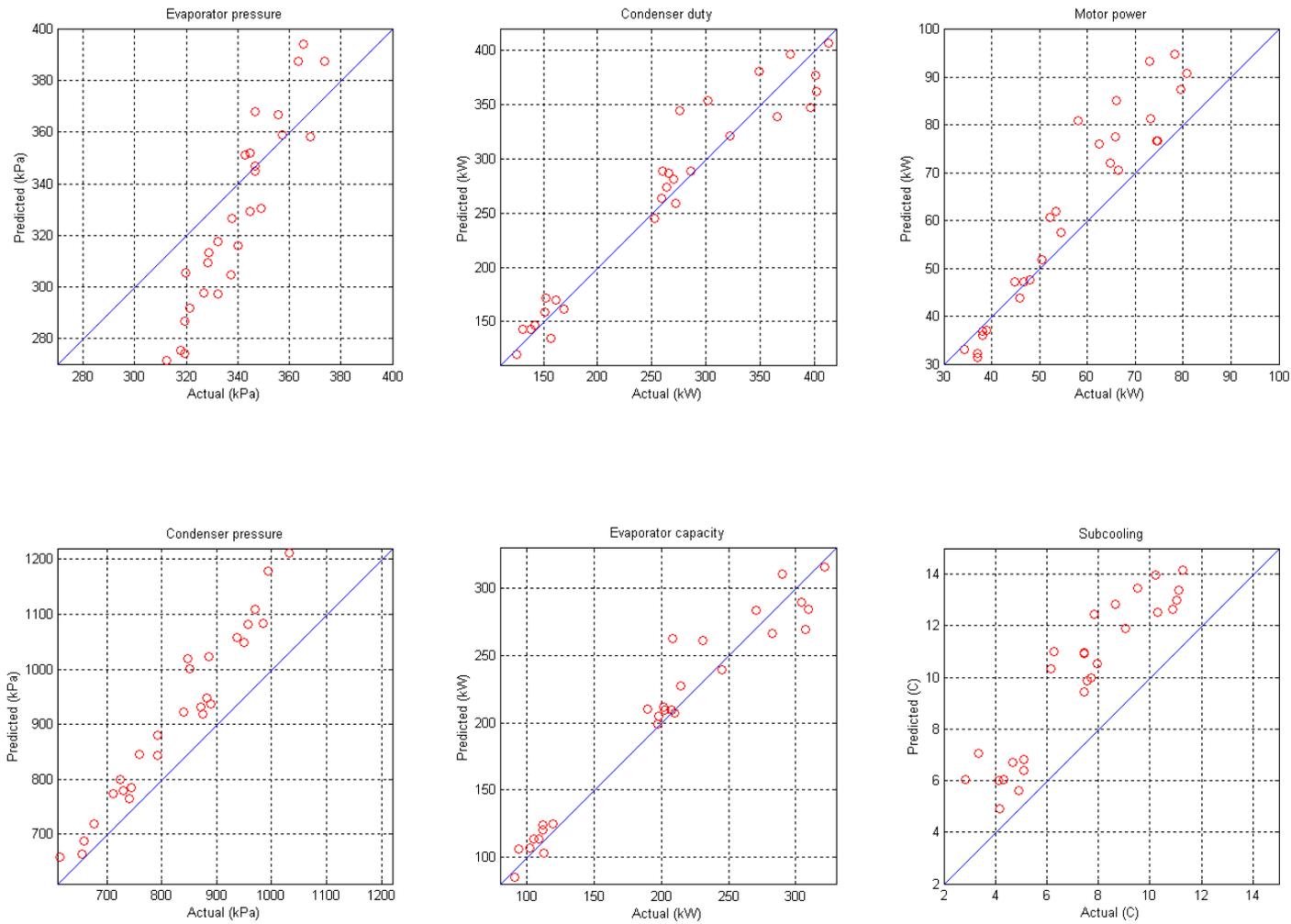


Fig. 25: Steady-state model performance with 20% excess refrigerant

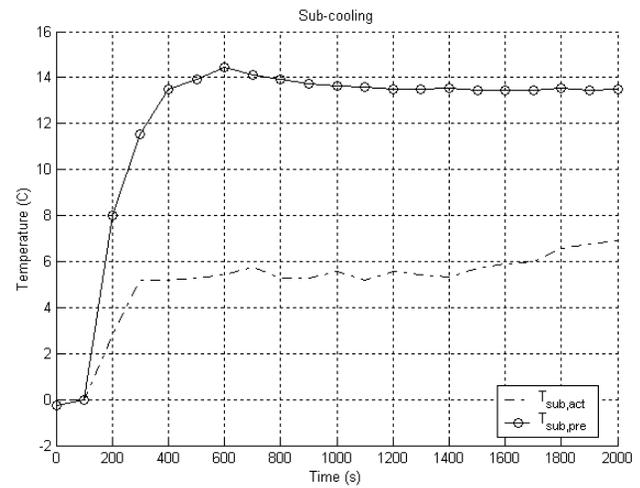
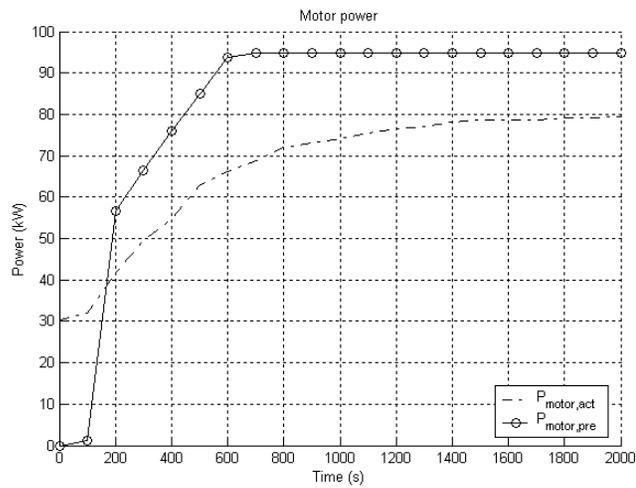
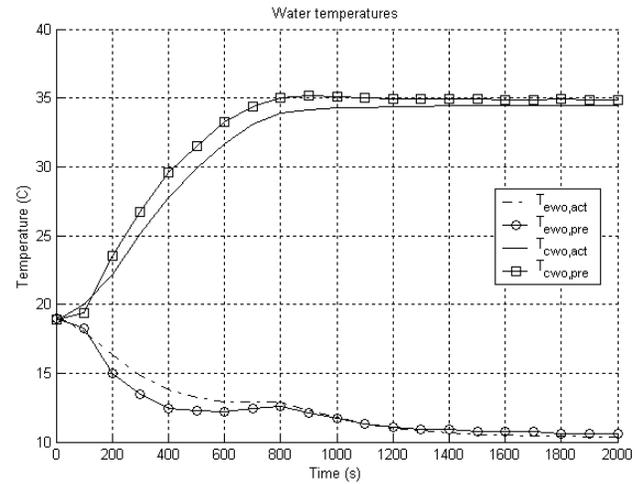
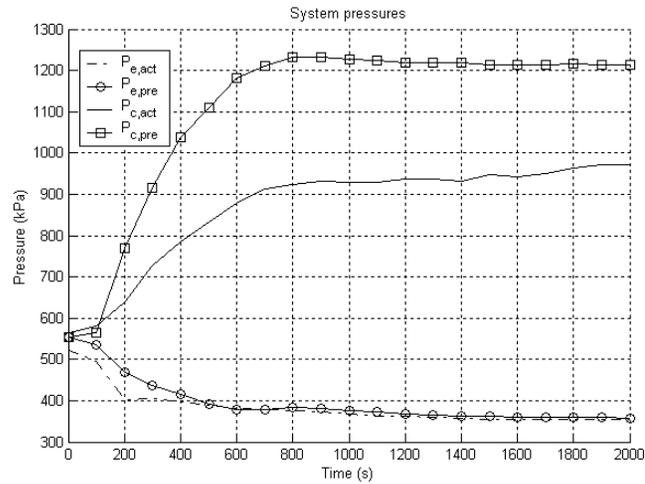


Fig. 26: Start-up model performance with 20% excess refrigerant

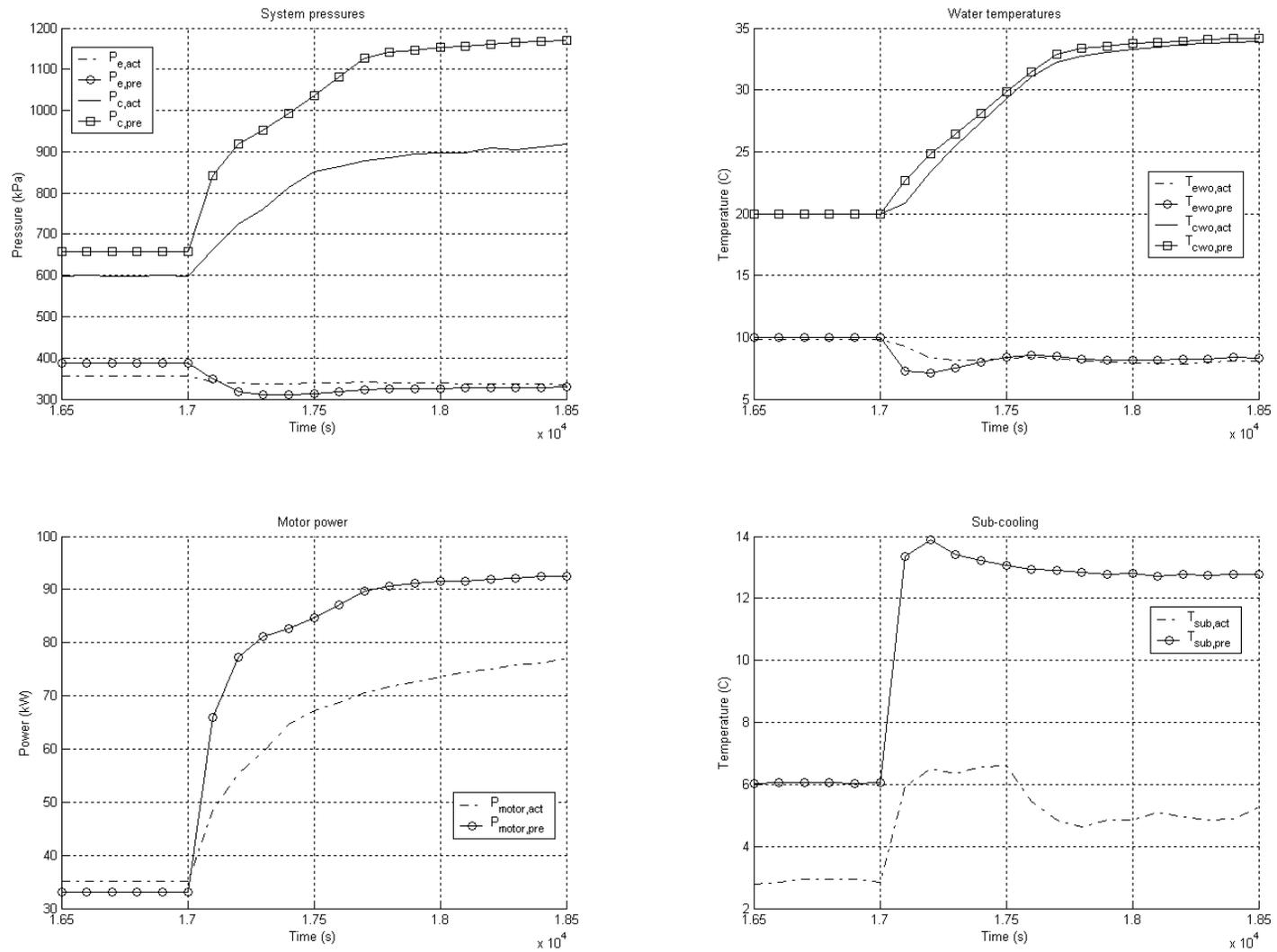


Fig. 27: Load change (LC9) model performance with 20% excess refrigerant

6.6 Condenser fouling

The final fault to be validated is condenser fouling. During data collection, this fault was implemented in the test set-up by physically plugging a pre-determined number of tubes in the condenser to prevent water from flowing through. This has the steady-state effect of reducing heat transfer area which is analogous to increased heat transfer resistance. The number of tubes plugged corresponded to the severity level of the fault, the severest being a 45% reduction in heat transfer area.

In the model, heat exchanger fouling can be more easily implemented by altering the heat-transfer coefficients. For the severest case of condenser fouling tested for, the water side heat transfer coefficient was scaled down to 55% of its normal value. Figs. 28, 29 & 30 show how this fault thus implemented in the model, compares with the measurements which were with plugged tubes. Table 7 compares the actual and predicted changes in system performance due to the fouling. Despite the differences in implementation between the system and the model, it is seen that the essence of the phenomenon has been captured in both cases and the model does a good job of predicting the steady-state and transient performance.

Table 7: Deviation of steady-states from fault-free with fouled condenser ation

SS No.	Evap. Pr (kPa)		Cond. Pr (kPa)		Cond. Duty (kW)		Evap. Cap. (kW)		Motor Power (kW)		Sub-cooling (C)	
	Act	Pre	Act	Pre	Act	Pre	Act	Pre	Act	Pre	Act	Pre
1	3.4	4.4	37.2	21.9	-26.3	3.9	-28.7	1.3	-2.1	1.5	-0.1	0.5
2	-7.6	-4.0	51.0	8.2	7.5	-16.3	-1.4	-17.0	3.9	-1.7	0.7	0.2
3	0.7	-2.5	20.7	22.7	-4.6	13.4	-6.1	10.4	-0.3	2.7	0.4	0.4
4	3.4	-1.6	57.9	26.1	-14.9	-6.4	-24.6	-7.3	1.4	1.4	0.3	0.4
5	-4.8	0.7	44.8	33.1	9.2	2.6	3.3	0.4	2.8	2.1	0.4	0.4
6	0.0	0.2	24.1	15.3	-0.6	1.9	-4.0	-0.1	0.4	0.7	0.4	0.2
7	-2.1	-3.8	48.3	49.0	0.9	24.8	-4.2	17.7	2.6	8.6	0.6	0.7
8	-3.4	-2.5	23.4	28.0	2.7	13.0	-2.6	10.6	1.6	2.7	0.2	0.5
9	-9.0	-0.1	27.6	18.6	10.1	2.8	5.3	0.5	1.7	2.4	0.5	0.3
10	3.4	-2.7	6.9	7.0	-24.8	-5.4	-28.0	-5.0	-2.8	-0.1	-0.4	0.4
11	-4.8	-13.8	26.9	32.8	-5.7	19.6	-8.9	13.7	1.1	5.8	0.4	0.5
12	0.7	3.3	6.9	3.1	-6.5	-11.6	-10.6	-11.6	-1.1	-1.9	-0.1	0.1
13	-1.4	1.2	48.3	24.7	-18.4	-4.5	-25.5	-4.4	1.7	1.2	0.6	0.4

Table 7 (Contd.): Deviation of steady-states from fault-free with fouled condenser

SS No.	Evap. Pr (kPa)		Cond. Pr (kPa)		Cond. Duty (kW)		Evap. Cap. (kW)		Motor Power (kW)		Sub-cooling (C)	
	Act	Pre	Act	Pre	Act	Pre	Act	Pre	Act	Pre	Act	Pre
14	1.4	1.2	17.2	9.4	-0.1	-5.3	-3.3	-4.7	0.1	-0.3	0.4	0.2
15	-0.7	-3.2	24.1	8.6	-19.3	-8.3	-19.9	-5.1	-0.5	-0.8	0.0	0.5
16	9.0	14.2	27.6	14.5	-14.6	-26.2	-16.9	-23.1	0.2	-3.0	0.0	0.1
17	-2.1	0.8	20.7	21.8	0.2	2.6	0.8	0.5	0.9	0.9	0.0	0.3
18	0.7	-2.0	20.7	18.0	-23.9	-10.3	-30.7	-8.1	-0.2	0.4	-1.3	0.4
19	-4.8	-2.7	24.1	39.8	-9.8	17.2	-11.9	11.6	0.8	6.0	-0.7	0.6
20	-1.4	1.4	13.8	6.8	-3.2	-6.7	-3.4	-5.9	-0.1	-0.6	-0.1	0.2
21	1.4	-1.4	37.9	13.0	-6.8	-5.9	-6.7	-6.4	2.8	0.4	-0.8	0.4
22	-4.1	1.8	20.7	12.7	-7.2	-3.0	-11.0	-3.5	0.1	0.2	-0.4	0.3
23	1.4	3.1	6.2	14.6	-4.7	-5.8	-9.0	-6.5	-0.4	-0.1	-0.3	0.2
24	-2.1	-2.8	-24.1	8.1	-36.9	-5.8	-40.6	-5.3	-6.3	0.2	-1.1	0.4
25	-2.1	-10.8	6.9	77.7	-14.9	56.0	-14.3	43.3	-0.9	11.9	-0.8	1.0
26	0.0	-3.4	0.0	44.3	-7.6	7.2	-9.7	5.3	-0.9	2.2	-0.4	0.3
27	1.4	1.2	17.2	9.4	-0.1	-5.3	-3.3	-4.7	0.1	-0.3	0.4	0.2

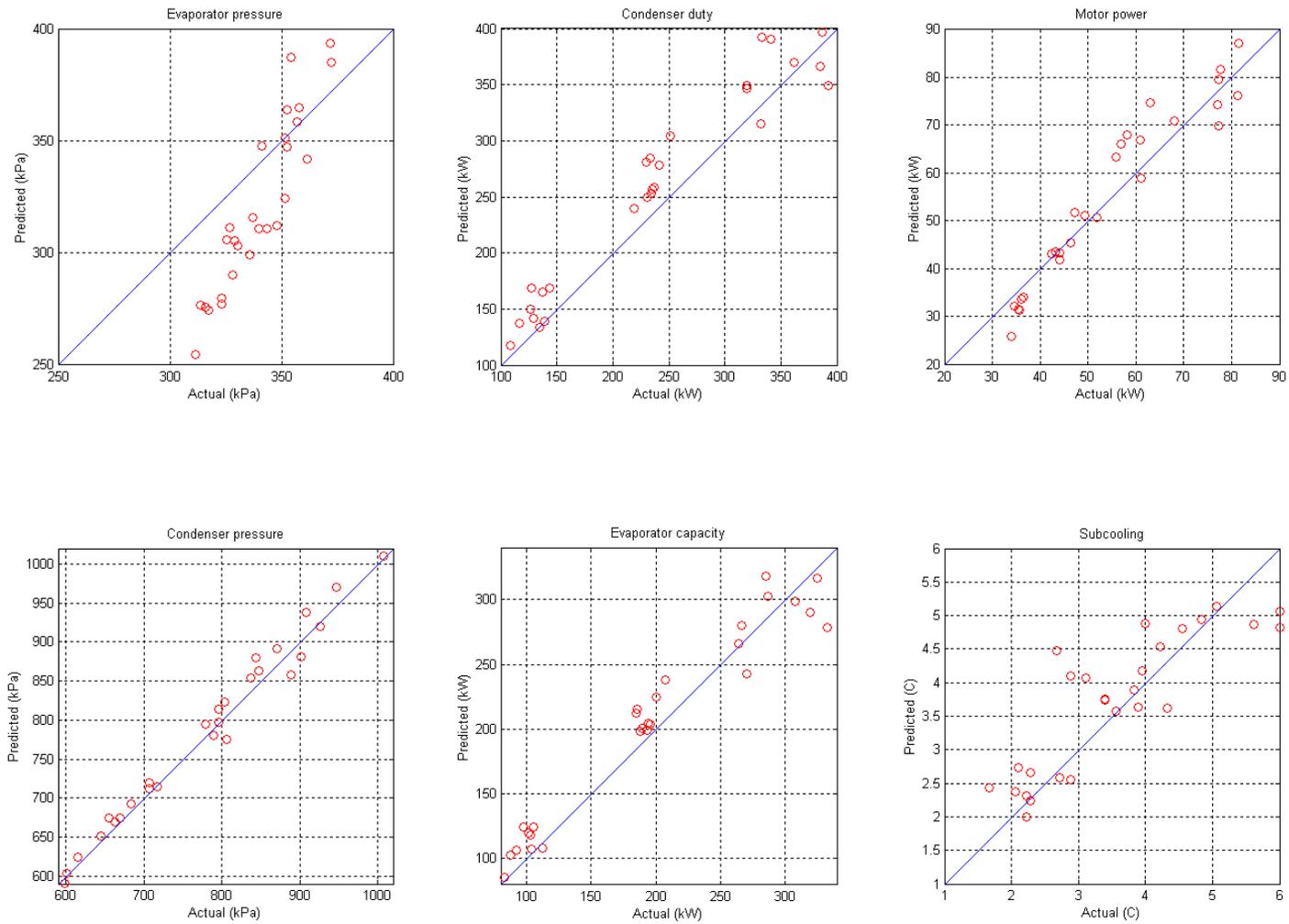


Fig. 28: Steady-state model performance with 45% fouling in condenser

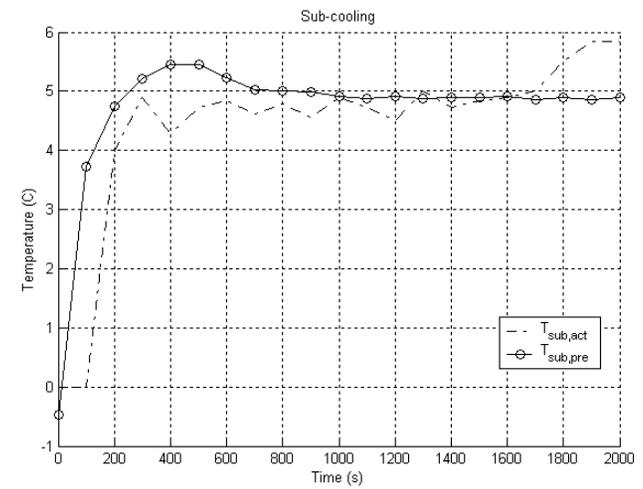
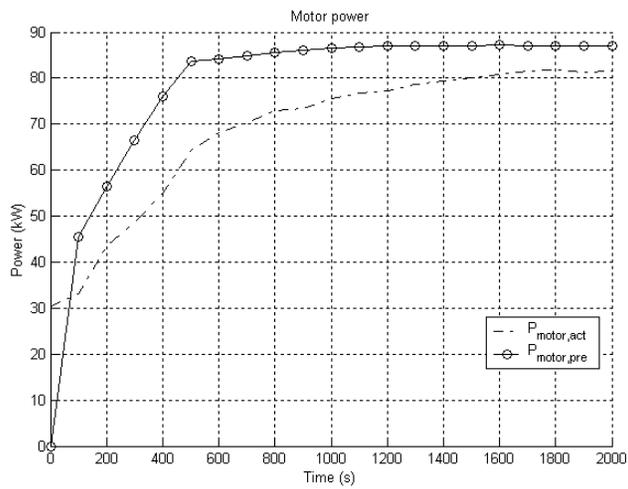
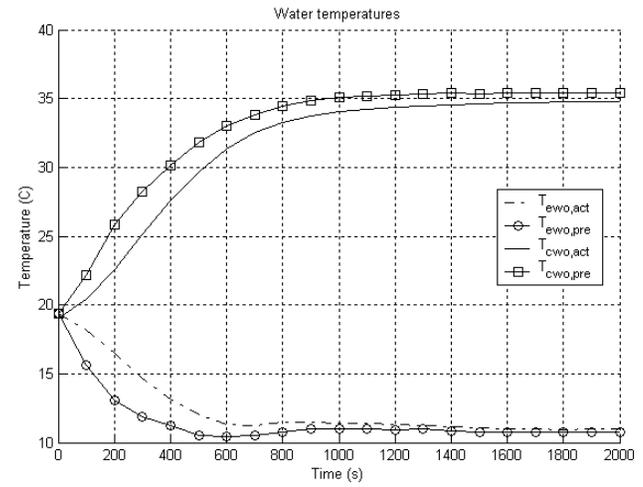
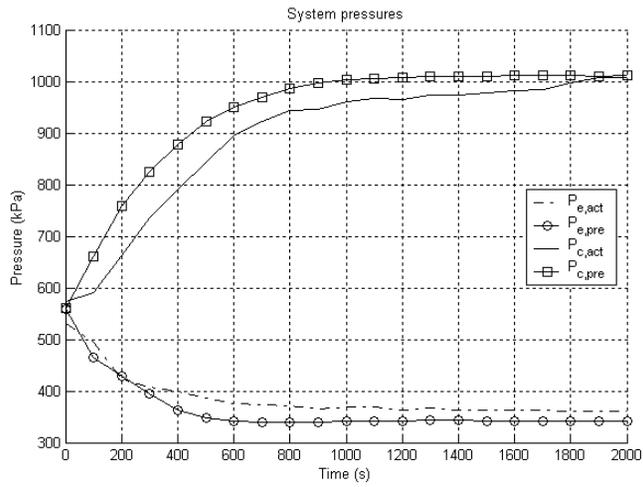


Fig. 29: Start-up model performance with 45% fouling in condenser

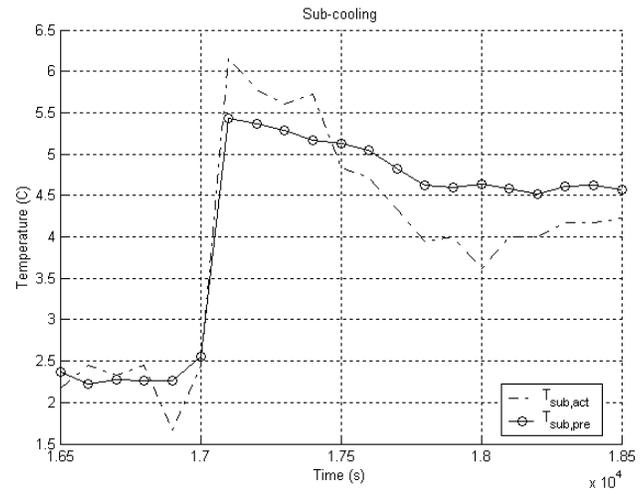
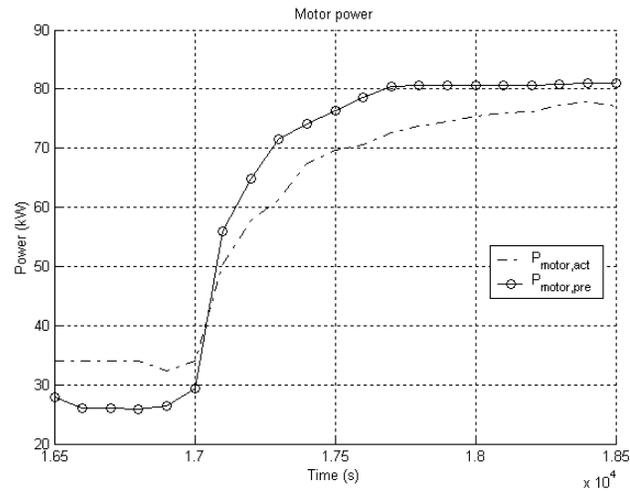
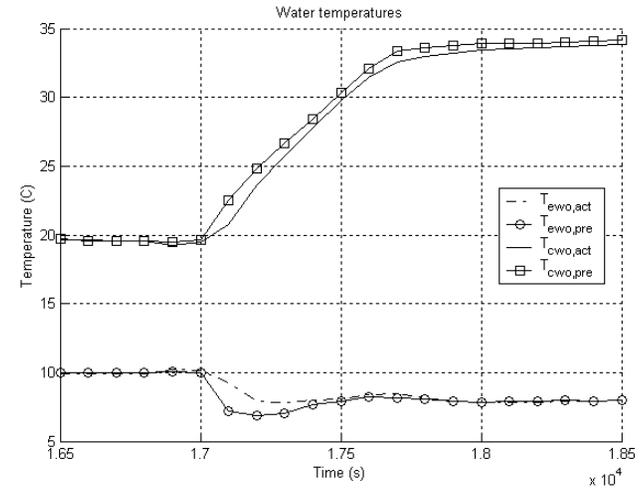
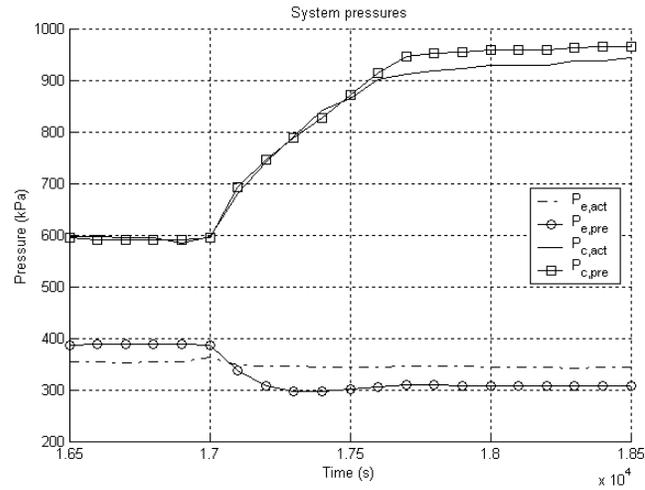


Fig. 30: Load change (LC9) model performance with 45% fouling in condenser

CONCLUSIONS

A dynamic model of a vapor compression centrifugal liquid chiller was developed from first principles. The model was validated using data from a 90-ton centrifugal chiller under a variety of transients, importantly, start-up and various load changes triggered by changes in evaporator return water temperature, condenser entering water temperature and chilled water set-point. The model predicts the measurements well, matching the important time-constants associated with the system. The model executes stably from start-up, through a sequence of 27 steady states with all the intermediate transients, until just short of shutdown. The execution speed, on a 500MHz/128MB computer, is slower than real-time by a factor of ~3. On a 1.8GHz/512MB computer, the execution time, as a ratio to real-time, is between 1.0-1.2.

In keeping with the objectives for which this model was developed, it was seen that feedback control functions well despite significant simplifications in the controller modeled. The chilled-water set point was reached well, and the system transitioned smoothly under all changes in supply water temperatures.

The model was also validated with faults introduced, specifically reduced condenser and evaporator water flow rates and refrigerant undercharge, and found to predict the measured performance well.

Deviations in the model predictions are observed in the form of the evaporator pressure at steady state. This is explained by the approximation used of a single thermostatic expansion valve in place of the pilot-valve arrangement and with which the non-linearities of the actual arrangement cannot be captured. However, in spite of this, the behavior of the other components indicates that the use of a more accurate expansion valve model (and possibly of a mechanistic implementation of the inlet-guide vane control of the centrifugal compressor) would result in an improved model performance. Over-prediction of condenser pressure when using the model in an overcharged condition, needs to be investigated further. Also required is an improvement in the correlation used for the condensing heat-transfer coefficient computation that will eliminate the use of a large correction factor.

The model can be used to study trends in system operating parameters under different conditions and can be expected to predict good response times and steady-state values. It must be remembered, however, that the system model is specific in that the compressor and its

associated controller are modeled from confidential design documentation made available from McQuay. Also specific to this model are the approximations related to the valve. These approximations were made using measured data and some parameter tuning. Although the methodology of the model can be adopted for other centrifugal chiller configurations, appropriate controller, compressor and valve models need to be developed for those systems. The heat exchanger models, on the other hand, are quite generic and can expectedly be used on any shell-tube constructions. Within the heat exchangers, the control volumes are required to encompass one or more tube-rows. That is, each control volume must contain integer number of tubes. Also, the control volumes must be aligned orthogonal to the refrigerant flow-direction. The adoption of single-water pass in place of two is seen not to impact the model performance noticeably.

The final specific of this system model is the use of water as the secondary fluid and R134a as the refrigerant. Use of other refrigerants will require the generation of property tables and calibration of the property routines. Appropriate changes also will be required to account for the heat-transfer correlations. Similarly, use of brine or water-glycol as the secondary fluid will require the development and calibration of property routines and heat-transfer coefficients.

It is expected that slowly developing faults, such as refrigerant leak and heat exchanger fouling can also be implemented in the system with minor changes to the system model. However, no such validation attempts have been made and this feature remains a model improvement that may be introduced into the model at a later date.

REFERENCES

- Bendapudi S., and Braun J.E., 2002, "A review of literature on dynamic models of vapor compression equipment", ASHRAE 1043-RP, TC 4.11.
- Braun J.E., Mitchell J.W. and Klein S.A., 1987, "Models for variable-speed centrifugal chillers", ASHRAE Transactions Vol. 93, Part 1, pp. 1794-1813.
- Breuker M.S, and Braun J.E., 1998, "Common faults and their impacts for rooftop air-conditioners", International Journal of HVAC&R Research, Vol. 4, NO. 3, pp. 303-318.
- Browne M. and Bansal P., 2000, "Modelling of in-situ liquid chillers", Proc. 2000 International Refrigeration Conference at Purdue, pp. 425-432.
- Browne M.W. and Bansal P.K., 1998, "Challenges in modeling vapor compression chillers", ASHRAE Transactions Vol. 104 Part 1, Paper No. 4141.
- Chi J. and Didion D.A., 1982, "A simulation model of the transient performance of a heat pump", International Journal of Refrigeration, Vol. 5, No. 3, pp. 176-184.
- Comstock M.C., 1999, "Development of analysis tools for the evaluation of fault detection and diagnostics in chillers", MSME Thesis, Purdue University.
- Dhar M. and Soedel W., 1979, "Transient analysis of a vapor compression refrigeration system.", XV International Congress of Refrigeration, Venice.
- Dhir V.K. and Lienhard J.H., 1971, "Laminar film condensation on plane and axisymmetric bodies in non-uniform gravity", Journal of Heat Transfer, Vol. 93, pg. 97.
- Dossat R.J., 1991, "Principles of refrigeration" 3rd Ed. (Chap. 12) Prentice Hall, Englewood Cliffs, NJ.
- Gnielinski V., 1976, "New equation for heat and mass transfer in turbulent pipe and channel flow." International Chemical Engineering, No. 16, pp. 359-368.
- Goldschmidt V.W. and Murphy W.E., 1979, "Transient performance of air conditioners", Proc. of Technical Groups, New Zealand Institute of Engineers, Vol. 5 (Issue 4, ISSN-0111-1358), pp. 715-738.
- Grace I.N. and Tassou S.A., 2000, "Dynamic simulation of liquid chillers", Proc. 2000 International Refrigeration Conference at Purdue, pp. 433-440.

- Grald E.W. and MacArthur J.W., 1992, "A moving boundary formulation for modeling time-dependant two-phase flows." *International Journal of Heat and Fluid Flow*, Vol. 13, No. 3, pp. 266-272.
- He X.D., Liu S and Asada H., 1994, "A moving interface model of two-phase flow heat exchanger dynamics for control of vapor compression cycle heat pump and refrigeration systems design, analysis and applications.", *AES Vol. 32*, ASME.
- He X.D., Liu S. and Asada H., 1986, "Modeling of vapor compression cycles for advanced controls in HVAC systems", *Proc. of the American Control Conference*, Seattle, Washington.
- Hoffman J.D., 1992, "Numerical methods for engineers and scientists", McGraw Hill, Inc.
- Incropera F.P. and Dewitt D.P., 1996, "Fundamentals of heat and mass transfer", 4th Ed, John Wiley and Sons.
- Jakobsen A, Antonius J. and Høgaard Knudsen H.J., 1999, "Experimental evaluation of the use of homogenous and slip-flow two-phase dynamic models in evaporator modeling", *Proc. 20th International Congress of Refrigeration*, Sydney. Paper No. 135.
- Kuo B.C., 1987, "Automatic control systems", Prentice Hall International Editions (Chap. 4)
- Mac Arthur J.W. and Grald E.W., 1989, "Unsteady compressible two-phase flow model for predicting cyclic heat pump performance and a comparison with experimental data", *International Journal of Refrigeration*, Vol. 12, pp. 29-41.
- MacArthur J.W. and Grald E.W., 1987, "Prediction of cyclic heat pump performance with a fully distributed model and a comparison with experimental data", *ASHRAE Transactions Vol. 93*, Part 2.
- MacArthur J.W., 1984, "Analytical representation of the transient energy interactions in vapor compression heat pumps.", *ASHRAE Paper, HT-84-19*, Vol. 90, Part 1.
- MacArthur J.W., 1984, "Theoretical analysis of the dynamic interactions of vapor compression heat pumps", *Energy Conservation Management*, Vol. 24, No. 1, pp. 49-66.
- MacArthur J.W., 1984, "Transient heat pump behavior: a theoretical investigation", *International Journal of Refrigeration*, Vol. 7, No. 2, pp. 123-132.
- Murphy W. and Goldschmidt V., 1985, "Cyclic characteristics of a typical residential air-conditioner – modeling of start-up transients", *ASHRAE Transactions*, Vol. 91, Part 2.

- Murphy W. and Goldschmidt V., 1984, "Transient response of air-conditioners – a qualitative interpretation through a sample case", ASHRAE Transactions, Vol. 90, Part 1.
- Murphy W.E., 1977, "Mobile home air-conditioning: an analysis of seasonal performance.", MS Thesis, School of Mechanical Engineering, Purdue University.
- Nusselt W., 1916, "Die Oberflächenkondensation des Wasserdampfes", Z. Ver. Deut. Ing., Vol. 60, pg. 541.
- Petukhov B.S., Irvine T.F. and Hartnett J.P., 1970, "Advances in Heat Transfer, Vol. 6, Academic Press, NY.
- Press W.H., Flannery B.P., Teukolsky S.A. and Vetterling W.T, 1991, "Numerical recipes in C", Cambridge University Press.
- Rasmussen B.D., 1997, "Variable speed hermetic reciprocating compressors for domestic refrigeration", PhD Thesis No. ET-PhD. 97-03, Department of Electrical Engineering, Technical University of Denmark, Copenhagen.
- Rasmussen R.W., Grald E.W., MacArthur J.W. and Ruohonieni T.J., 1986 "Gas heat pump system and component efficiency and reliability improvement development: controls and valves", Final Report to Gas Research Institute.
- Rossi T.M and Braun J.E., 1999, "A real-time transient model for air conditioners", Proc. 20th International Congress of Refrigeration, Sydney. Paper No. 743.
- Sami S.M., Duong T., Mercadier Y and Galanis N., 1987, "Prediction of the transient response of heat pumps", ASHRAE Transactions, Vol. 93, pg. 471.
- Svensson M.C., 1999, "Non-steady-state modeling of a water-to-water heat pump unit", Proc. 20th International Congress of Refrigeration, Sydney, Paper No. 263.
- Svensson M.C., 1994, "Studies on on-line optimizing control with application to a heat pump", PhD Thesis, The University of Trondheim.
- Tillner-Roth R. and Baehr H.D., 1994, "An International Standard Formulation for the Thermodynamic Properties of 1,1,1,2-Tetrafluoroethane (HFC-134a) for Temperatures from 170 K to 455 K and Pressures up to 70 MPa", Journal of Physical Chemistry, Ref. Data, Vol. 23, No. 5.
- Wang H. and Wang S., 2000, "A mechanistic model of a centrifugal chiller to study hvac dynamics", Building Services Engineering Research and Technology, Volume 21(2), pp. 73-83.

- Wang H. and Touber S., 1991, "Distributed and non-steady-state modeling of an air-cooler", International Journal of Refrigeration, Vol. 12.
- Wedekind G.L., Bhatt B.L. and Beck B.T., 1978, "A system mean void fraction model for predicting various transient phenomena associated with two phase evaporating and condensing flows.", International Journal of Multiphase Flow, Vol. 4, pp. 97-114.
- Xiangdong He, Sheng Liu and Asada H.H., 1998, "Multivariable control of vapor compression systems", International Journal of HVAC&R Research, ASHRAE July 1998, Vol.4, No.3.
- Xiandong He, Sheng Liu and Asada H.H., 1997, "Modeling of vapor compression cycles for multivariable feedback control of HVAC systems", ASME Journal of Dynamic Systems, Measurement and Control, Vol. 119, No. 2, June 1997.
- Yasuda Y. Touber S and Machielson, 1982, "Simulation model of a vapor compression refrigeration system", ASHRAE Transactions Paper No. 2787, pp. 408-425.

APPENDICES

APPENDIX A: Linearization of the conservation equations

Beginning with the mass balance equation on the i^{th} node:

$$\frac{d(\rho V)_i}{dt} = \dot{m}_{i-1} - \dot{m}_i \quad (34)$$

For any node, the volume V_i , is a constant, and the density can be expressed as a function of the two independent variables of pressure, P and enthalpy h, as:

$$\frac{d\rho}{dt} = \frac{\partial\rho}{\partial P} \frac{dP}{dt} + \frac{\partial\rho}{\partial h} \frac{dh}{dt} \quad (35)$$

Defining properties a_i and b_i as:

$$a_i = V_i \left(\frac{\partial\rho}{\partial P} \right)_h \quad b_i = V_i \left(\frac{\partial\rho}{\partial h} \right)_P \quad (36)$$

the mass balance equation becomes:

$$a_i \frac{dP}{dt} + b_i \frac{dh}{dt} = \dot{m}_{i-1} - \dot{m}_i \quad (37)$$

In the energy equation given by:

$$\frac{d(\rho u V)_i}{dt} = \dot{m}_{i-1} h_{i-1} - \dot{m}_i h_i - \dot{Q}_{r,i} \quad (38)$$

the internal energy is replaced by the enthalpy, using the following substitution:

$$u = h - Pv \quad \Rightarrow \quad \rho u = \rho h - P \quad (39)$$

differentiation of which, yields:

$$\frac{d(\rho u)}{dt} = \rho \frac{dh}{dt} + h \frac{d\rho}{dt} - \frac{dP}{dt} \quad (40)$$

Substituting eqn. (35) into eqn. (40):

$$\frac{d(\rho u)}{dt} = \left(h \frac{\partial\rho}{\partial P} - 1 \right) \frac{dP}{dt} + \left(h \frac{\partial\rho}{\partial h} + \rho \right) \frac{dh}{dt} \quad (41)$$

Defining properties c_i and d_i as:

$$c_i = V_i \left[h \left(\frac{\partial \rho}{\partial P} \right)_h - 1 \right] \quad d_i = V_i \left[h \left(\frac{\partial \rho}{\partial h} \right)_P + \rho \right] \quad (42)$$

the energy balance becomes:

$$c_i \frac{dP}{dt} + d_i \frac{dh}{dt} = \dot{m}_{i-1} h_{i-1} - \dot{m}_i h_i - \dot{Q}_{r,i} \quad (43)$$

APPENDIX B: Property evaluations

Refrigerant and water properties used in this model are computed from routines developed specifically for this application. Using *EES* (Engineering Equation Solver), a set of high-accuracy, thermo-physical and transport property look-up tables were generated for the three phases of the refrigerant, namely, sub-cooled, two-phase and superheated. The properties used in *EES* are based on the fundamental equation of state developed by Tillner-Roth and Baehr [1994]. The refrigerant enthalpy is referenced to 200kJ/kg at 273.15K. These tables were generated over a pressure range from 10kPa to 3500kPa, and from a sub-cooled condition of about 25°C to a superheated condition of about 50°C. Similarly, transport properties of water were generated for a range of temperatures. For the R500 used in the sensing bulb, only two-phase properties were generated. The text files associated with the properties are tabulated below:

Table 8: Property tables and associated text files

File Name	Description
<i>r134ahatpth.txt</i>	Two-phase, thermo-physical properties of R134a
<i>r134ahatptr.txt</i>	Two-phase, transport properties of R134a
<i>r134ahascth.txt</i>	Sub-cooled, thermo-physical properties of R134a
<i>r134asctr.txt</i>	Sub-cooled, transport properties of R134a
<i>r134ashth.txt</i>	Superheated, thermo-physical properties of R134a
<i>r134ashtr.txt</i>	Superheated, transport properties of R134a
<i>waterprops.txt</i>	Liquid water properties
<i>r500tpth.txt</i>	Two-phase properties of R500

During model execution, the very first process is the loading of property tables into memory. This process consists of preparing global, read-only matrices from the look-up tables. There exist a set of 8 such global matrices, namely, REF2PTH, REF2PTR, WATERLQ, BULBPTH, REFSCTH, REFSCTR, REFSHTH and REFSHTR. Next, for the single-phase property tables, splines are fitted for each property, as a function of the enthalpy. This is a one-time computation that is done at the time of loading the properties into memory. As constructed, the property tables are used to extract complete state information using pressure and enthalpy as the independent variables. In the single-phase regions, at run-time, bi-cubic spline interpolation is used to evaluate the other properties. In the two-phase regions

(for either R134a or R500) and for water, this can be achieved by a simple spline interpolation on the pressure, followed by linear interpolation using the quality.

Given a state (P,h) , the two phase properties are computed first, based on the pressure. From the saturated liquid and vapor enthalpies at this pressure, the phase is determined. If it is a two-phase condition, the quality is computed and all relevant properties are interpolated linearly between the saturated conditions. If it is a single-phase condition, bi-cubic spline interpolation is applied to the appropriate table, and all relevant properties are obtained.

The operative entity in the property evaluation is the class *r134astate*. This class has a method *setstate(P,h)*, that performs the above tasks for given pressure and enthalpy. It was found convenient to compute all properties, thermo-physical as well as transport, in one call of this method, as opposed to computing one specific property at a time. Although this might appear wasteful, it is justified as follows. During system simulation, the most frequent use of property evaluations occurs in the heat exchangers, when computing the nodal properties. Within each heat exchanger, the properties of each node need to be computed, once in the predictor step, and once in the corrector step. And at this point, all the major refrigerant properties, i.e. sp. volume, internal energy, all transport properties, are required. This prompted the one-step all-property computation choice.

The properties used in this model were calibrated against *EES* under a range of states and found to be of high-fidelity. The numerical routines for spline generation and interpolation were taken from Press et al [1991].

APPENDIX C: Valve geometry approximation

The unavailability of detailed information about the expansion valve arrangement's construction necessitated the use of a simplified expansion valve. Assumptions were made regarding the geometry of the needle-valve and spring, and these dimensions were tuned during system simulation to prevent choking, or flooding.

The expansion device is assumed to consist of a circular seat, operated by a conical valve that is acted on by a spring, as shown in the sketch below.

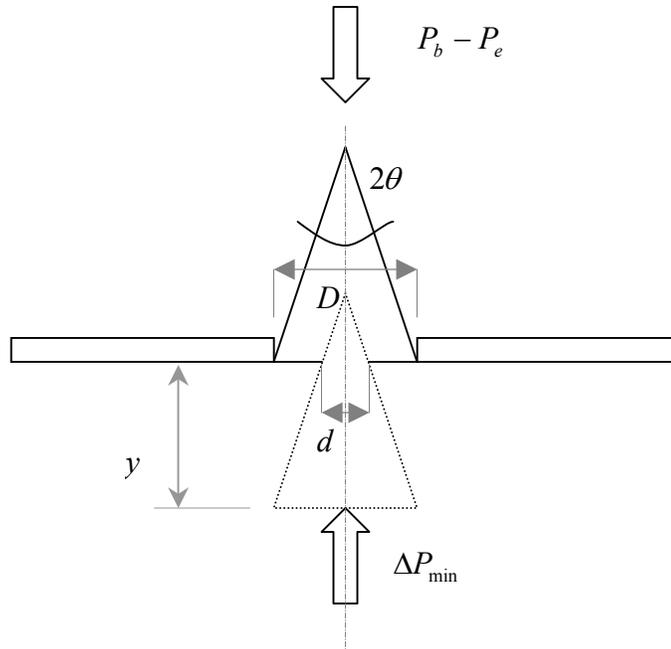


Fig. 31: Approximated geometry used in valve model

For given P_b and P_e , y is determined from eqn. (28). At any y , the annular flow area is given by

$$A_v = \pi(D^2 - d^2)/4 \quad (44)$$

where

$$d = (D - 2y \tan \theta) \quad (45)$$

Substituting (44) into (43) and re-grouping terms in y and y^2 gives the flow area in the form of (29). Using a value of 20° for θ , the value of D was computed by setting the maximum flow area at 250 mm^2 . Table 9 summarizes all the assumed and computed values used in the valve model.

Table 9: Valve parameters

Parameter	Value	Units
A_{max}	250	mm^2
θ	20	deg
D	17.8	mm
ΔP_{min}	80	kPa
k_{spring}	42e-5	kPa/m
a_0	0.020400	m
a_1	-0.416180	ND
C_b	100	$\text{kW}/^\circ\text{C}$

Tuning this model of the valve involves some amount of trial-and-error coupled with a parametric study of the system performance. However, the modular nature of the model allows for incorporation of alternate models (or maps) that meet the requirements of the information flow diagrams. Such attempts are deferred to the future.

APPENDIX D: Model software user manual

The chiller model documented in section 4 of this report was coded in C++. The property routines were developed as described in Appendix B. This appendix begins with a description of all the files that are a part of the software. This is followed by a description of how the model is executed from within Matlab. Finally, the process for simulating the more common faults in the system model is described.

Installation and file/directory structure:

The complete system model is packaged as a compressed (zip) file named *chillersim1p0.zip*. Installation consists of un-compressing this file to a known directory and including this directory and all sub-directories therein into Matlab's search path.

Fig. 32 shows the file and directory structure seen upon uncompressing *chillersim1p0.zip*:

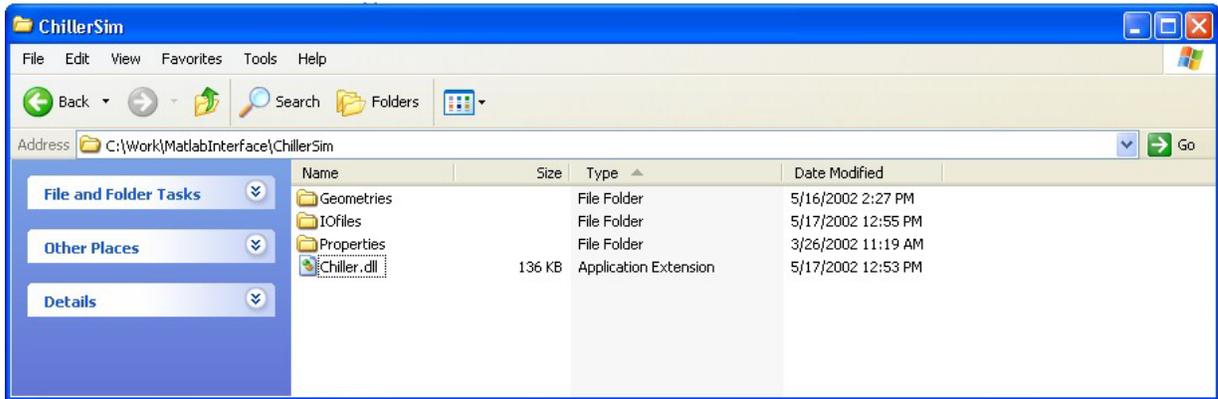


Fig. 32: Screen-shot of unzipped chiller model files

Chiller.dll is the dynamically linked library containing all the routines required to run the chiller components and system. The *Geometries* sub-directory contains the text files with the physical constructional details of the various components of the chiller. The *IOFiles* sub-directory contains the text files that are used by the chiller model during initialization and execution. The *Properties* sub-directory contains the property tables that are read into memory when the model is first launched.

Fig. 33 shows the text files in the *Geometries* directory.

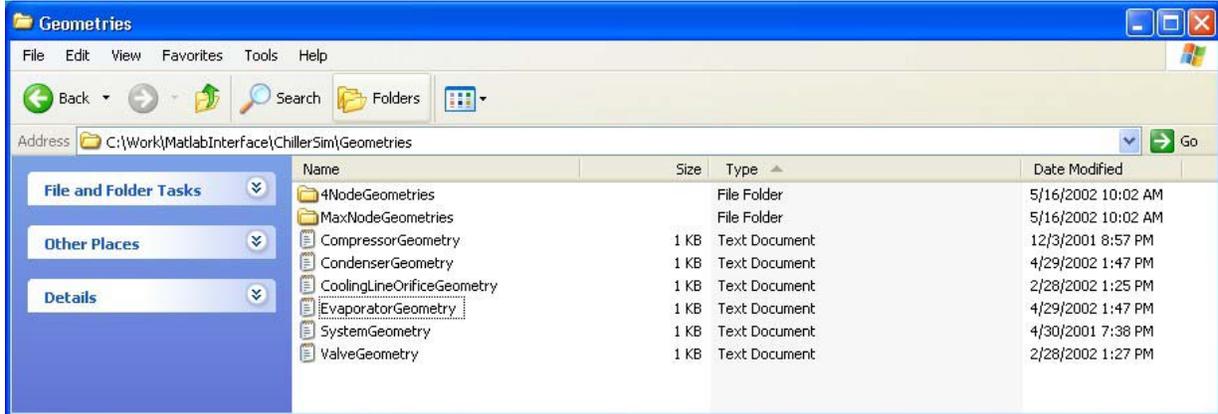


Fig. 33: Screen-shot of the Geometries directory

COMPRESSORGEOMETRY.TXT: This file contains the details required for defining the controller and compressor models. It is strongly recommended that these values not be changed as they could result in unpredictable behavior of the compressor model and hence the system model. Appended at the end of the file is a line-by-line description of parameters.

VALVEGEOMETRY.TXT: This file identifies and lists the constructional parameters used in the valve model. This information consists of (also see Appendix C) the maximum flow-area of the valve, the angle of the valve needle, the discharge coefficient, spring compliance, sensing bulb time constant and the minimum superheat pressure setting.

COOLINGLINEORIFICEGEOMETRY.TXT: This file identifies and lists the required constructional parameters used to define the orifice in the cooling line which is the flow-area of the orifice and the discharge coefficient.

CONDENSERGEOMETRY.TXT, EVAPORATORGEOMETRY.TXT: Since the evaporator and condenser are based on the same model, the constructional information required to define either one is the same and both of these text files are structured identically.

Referring to Fig. 34, the first line consists of two integer fields. The first integer identifies whether the data that follows is for an evaporator (a value of '1') or for a condenser (a value of '2'). The second integer identifies the number of nodes that the heat exchanger is discretized into.

```

CondenserGeometry - WordPad
File Edit View Insert Format Help
[Toolbar icons]
2      4
0.01554    0.01905    2.4384    0.385 8933  0.45
58      0.053664252927  0.1421
62      0.042021948429  0.0934
25      0.022416845907  0.0467
19      0.040977406716  0.0954

Line 1:    Heat exchanger type (1=evaporator, 2=condenser)    Number of nodes
Line 2:    Tube inside diameter (m), Tube outside diameter (m), Tube length (m), Tube material Cp (kJ/kg-K), Tube material density (kg/m^3)
Line 3-end: Nodal geometry according to following table

No. of tubes in node    Refrigerant volume of node (m^3)    Vertical height of node (m)
1.
2.
.
.
.
NNodes

The nodes are numbered with 1 being the node into which refrigerant enters the heat exchanger, and proceeding sequentially to NNodes out of which the refrigerant leaves the heat exchanger.

For Help, press F1
NUM

```

Fig. 34: Heat exchanger geometry specification format

The second line contains information about the tube size and material, listed in the order of inside diameter, outside diameter, length, specific heat and density. The sixth entry in this line is a fouling factor that will be described in the section on fault implementation.

All subsequent lines provide node specific information. The total number of lines must correspond to the number of nodes specified as the second integer in the first line. A mismatch will result in an error message and program termination. The lines are also to be arranged in the order that corresponds to the nominal flow-direction of the refrigerant through this heat exchanger with the first node being the one into which the refrigerant enters the heat exchanger and the last node as the one from which it leaves.

Each line of node information consists of three fields. The first field is the integer number of tubes encompassed in that node. The second field is the refrigerant volume in that node. The third field is the distance between the node faces in the vertical direction, i.e. in the direction of refrigerant flow.

SYSTEMGEOMETRY.TXT: This file lists the paths of all the other geometry text files and can be used to load different component details located in different directories.

The sub-directories *4NodeGeometries* and *MaxNodeGeometries* contain ready to use condenser and evaporator geometry files. The former apply to each heat exchanger being discretized into 4 nodes. The latter apply to each heat exchanger being discretized into as

many nodes as there are tube-rows in that heat exchanger (12 in case of the evaporator and 13 in case of the condenser).

The *IOFiles* sub-directory is where the input to and output from the model are stored. Fig. 35 shows the files within this sub-directory. The information required to initialize the system for start-up is stored in the text file(s) *Initial_*.txt*. The two possible initialization modes are designated *FULL* and *MINIMAL*. Full initialization consists of specifying the refrigerant pressure and refrigerant enthalpy⁵ in both heat exchangers. Minimal initialization requires only three values, i.e., water temperatures leaving the evaporator and condenser, and total refrigerant charge in the system. It is to be noted that both of these modes of initialization pre-suppose that the system is in an equilibrium condition corresponding to the instant before start-up. This means that whichever mode is used to initialize the system, the controller begins with the compressor's inlet guide vanes in the minimum opening position and the RLA limit at its minimum.

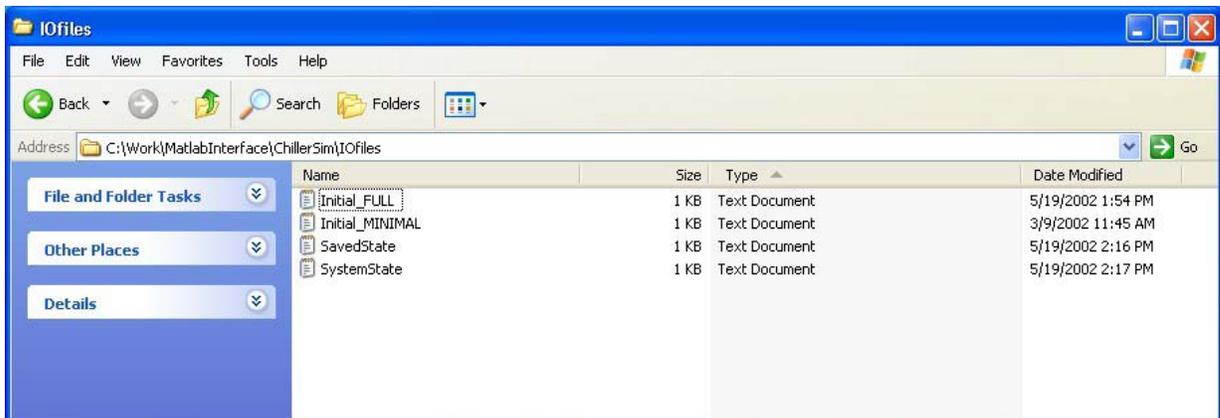


Fig. 35: Screen-shot of IOFiles directory

Once the chiller execution is begun, the controller gradually ramps up the limiting motor power thereby modeling the soft-start motor protection feature of the physical controller. In other words, the chiller system model can only be executed in a way that begins with a start-up. At the end of every 1s of simulation time, (i.e. the end of every loop of the second nested level described in section 5.1) the state of the system is saved in the text file *SystemState.txt*.

⁵ Please see Appendix B for more information on refrigerant properties. The refrigerant enthalpy is referenced to 200 kJ/kg at 273.15K and the specific entropy to 1.00 kJ/kg-K at 273.15K, with the refrigerant in saturated liquid condition.

There may exist a text file by name *SavedState.txt*. This file is created by the program when the user requires the current state to be saved between sessions. Saving and restoring the chiller state between sessions is described under the section on Usage. The following are the descriptions of the text files in the *IOFiles* directory.

INITIAL_FULL.TXT: This is the file read by the chiller model when a full initialization is required. The format of information in this file is as shown in Fig. 36.

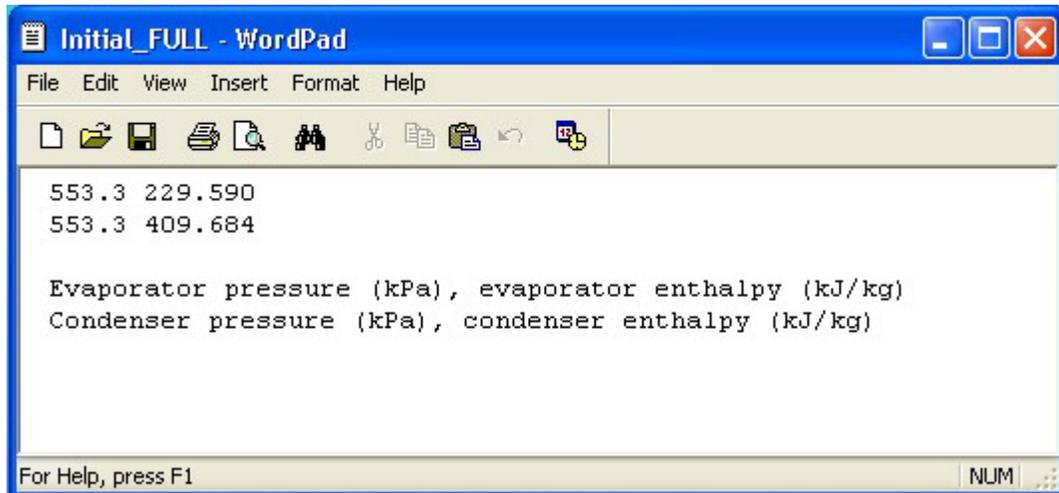


Fig. 36: Screen-shot of text file format for full initialization

INITIAL_MINIMAL.TXT: This is the file read by the chiller model when a minimal initialization is required and is shown in Fig. 37.

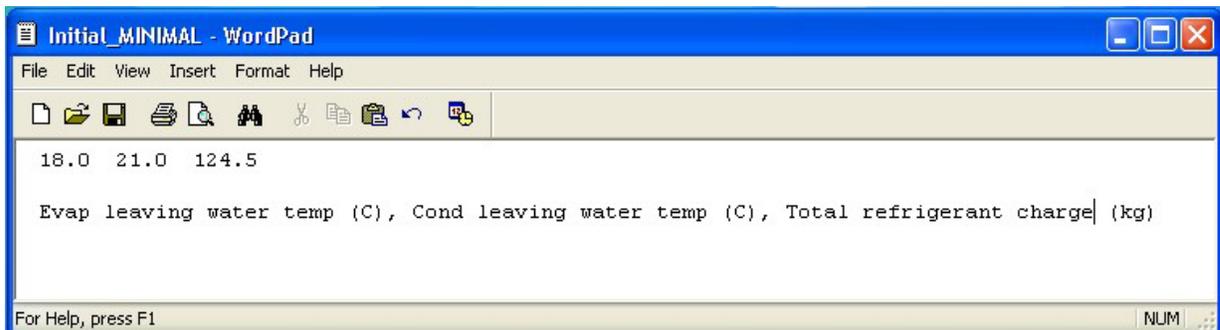


Fig. 37: Screen-shot of text file format for minimal initialization

SYSTEMSTATE.TXT: The system's states are saved in this file automatically every 1s of simulation in the format shown in Fig. 38.

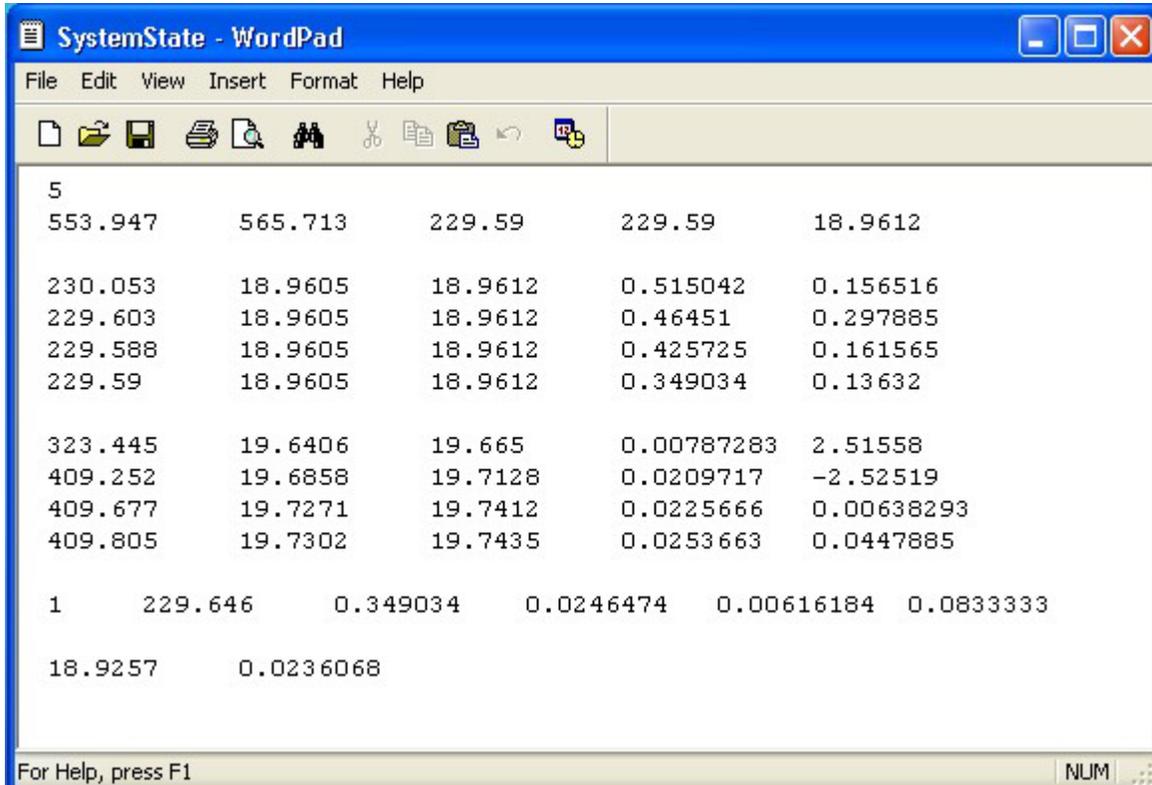


Fig. 38: Screen-shot of text file format for system states information

Line1: simulation time i.e. number of seconds that the model has run since start-up

Line 2: evaporator pressure, condenser pressure, evaporator exit enthalpy, condenser exit enthalpy, chilled water temperature.

Evaporator nodal states (as many lines as there are nodes in the evaporator) in the following order:

nodal refrigerant enthalpy in kJ/kg;

nodal tube temperature in °C ;

nodal water temperature in °C ;

nodal refrigerant-side heat transfer rate in kW and

nodal refrigerant mass flow rate in kg/s.

Condenser nodal states (as many lines as there are nodes in the condenser) in the same format as above for the evaporator nodal states

Compressor state in the following order:

controller operation mode (1 = start-up, 2 = normal);
exit enthalpy in kJ/kg;
mass flow rate in kg/s;
motor power in kW;
motor heat losses in kW and
normalized guide-vane position (γ).

Valve and bulb state in the following order:

bulb temperature in °C and
mass flow rate in kg/s.

The *SavedState.txt* file (if one exists) has a format identical to the *SystemState.txt* file above. The text files in the *Properties* directory are as described in Appendix B.

Usage:

The chiller function can be called in Matlab with either one argument on the right-hand-side and none on the left, or with two arguments on the right-hand-side and two on the left, as below:

chiller(n)	(Single argument call)
y = chiller(t,u)	(Two-argument call)

Single argument call:

When called with a single, integer argument the following actions are taken depending on the value of the integer argument:

chiller(0) – performs a minimal initialization reading from *Initial_MINIMAL.txt*.

chiller(1) – performs a full initialization reading from *Initial_FULLL.txt*

chiller(2) – saves the current state of the system to *SavedState.txt*.

chiller(3) – loads the state of the system saved in *SavedState.txt*.

If a repeat initialization call is made, the existing chiller state information is simply overwritten. It is very important to note that all text filenames used by the program are unique and are overwritten without warning. Therefore, if any of the text files are desired by the user to remain unchanged, such files must be either renamed or relocated into another directory. This apparent lack of user-friendliness in fact allows for greater flexibility by allowing incorporation of the model into a program as a function call that is self-contained and that requires no real-time inputs from the user.

The chiller can also be initialized by reloading state information saved from an earlier session. This is done by entering 'chiller(3)' at the Matlab command prompt. On executing this, the text file *SavedState.txt* in the *IOFiles* directory is read and the system is restored to the state that existed when this text file was created. This file is created automatically by the program upon entering 'chiller(2)' at the command prompt. Any existing *SavedState.txt* file will be overwritten.

Two-argument-call:

After successful initialization, the chiller model can be executed by entering the following:

$$y = \text{chiller}(t,u)$$

't' and 'u' are the inputs required to drive the chiller model and 'y' is the output returned. The following is a description of these parameters.

t is an integer, positive number of seconds that the chiller is to be run;

u is the (5 x 1) vector of water-side boundary conditions in the order:

u[1] = Evaporator water entering temperature in °C

u[2] = Condenser water entering temperature in °C

u[3] = Chilled water set point temperature in °C

u[4] = Evaporator water mass flow-rate in kg/s

u[5] = Condenser water mass flow-rate in kg/s

y is a (29 x 1) vector of various system performance outputs, in the following order:

y[1] = Chiller simulation time since start-up in s

y[2] = Evaporator pressure in kPa

y[3] = Condenser pressure in kPa

- y[4] = Refrigerant flow rate through compressor in kg/s
- y[5] = Refrigerant flow rate through valve only in kg/s
- y[6] = Refrigerant flow rate through cooling line only in kg/s
- y[7] = Sum of y[5] and y[6]
- y[8] = Motor power in kW
- y[9] = Motor heat losses in kW
- y[10] = Condenser water-side heat transfer rate in kW
- y[11] = Evaporator water-side heat transfer rate in kW
- y[12] = Evaporator leaving water temperature (chilled water temperature) in °C
- y[13] = Condenser leaving water temperature in °C
- y[14] = Superheat in °C
- y[15] = Sub-cooling in °C
- y[16] = Condenser refrigerant mass imbalance in kg
- y[17] = Evaporator refrigerant mass imbalance in kg
- y[18] = Energy balance across compressor in kW
- y[19] = Energy balance across condenser in kW
- y[20] = Energy balance across evaporator in kW
- y[21] = Refrigerant specific enthalpy leaving evaporator in kJ/kg
- y[22] = Refrigerant specific enthalpy leaving compressor in kJ/kg
- y[23] = Refrigerant specific enthalpy leaving condenser in kJ/kg
- y[24] = Refrigerant specific enthalpy entering evaporator in kJ/kg
- y[25] = Valve lift in m
- y[26] = Valve flow area in m²
- y[27] = Refrigerant mass in condenser in kg
- y[28] = Refrigerant mass in evaporator in kg
- y[29] = Total refrigerant mass in the system in kg

The usage of the model is illustrated by a series of examples described below and included as m-files with the software. These m-files can be used as templates by the user.

Example 1 (*Ex1.m*): Start-up in fault-free condition:

This example demonstrates the preparatory steps that precede execution of the model, followed by the actual execution of the chiller model through the start-up. When the steady-state is reached, the execution is stopped and the state of the system at that time is saved for future use.

Step 1- System Definition: The default heat-exchanger geometry is used, i.e., as defined in the files in the *Geometries* directory.

Step 2 – Initialization: The default initialization of *Initial_FULLL.txt* is used.

Step 3 – Boundary conditions: The u vector is defined for the start-up period. For simplicity, it is assumed to remain constant during the complete start-up region. The desired set point is 10°C. The normal water-flow rates of 13.2kg/s in the evaporator loop and 16.7kg/s in the condenser water loop are used. The evaporator return water temperature is 16°C and the condenser return water temperature is 30°C.

Referring to the code in *Ex1.m*, (Fig. 39) line 4 is the *FULL* initialization step. This is followed by the setting of the water-side boundary conditions. The chiller output plotting rate is specified in line 29. With this information, the chiller execution loop is begun at line 34. For certain combinations of initial conditions and entering water temperature change rates during early (<150s) start up, it has been found that the model fails to converge. A full characterization of this numerical issue is in progress, but it can be overcome by gradually and linearly ramping the entering water temperature from the initial condition to the final value over the first 120s-150s. This is shown in lines 39-43.

Line 44 updates the input vector u and is followed by the execution of the chiller through a 10s loop. The output of the chiller is recorded at every 1s and saved in the *output* array which is saved to the disk every 10s (line 57). The 0.1s pause at line 59 is required only to allow the figure plots to refresh. The final state of the chiller is saved (line 62) into *SavedState.txt* for future use. Fig. 40 shows selected output of this example.

```
c:\work\MatlabInterface\ChillerSim\Ex1.m
File Edit View Text Debug Breakpoints Web Window Help
Stack: Base
1 %Chiller Model Examples
2 %Example 1
3 %
4 chiller(1);
5
6 %Define water-side boundary conditions
7 TEWI = 16;
8 TCWI = 30;
9 TEWO_SET = 10;
10 MEWAT = 13.2;
11 MCWAT = 16.7;
12
13 %Set-up plotting
14 FIG = figure;
15 set(FIG,'Position',[231 132 1128 908]);
16 subplot(311); axis([0 1000 0 1500]);
17 grid on; hold on;
18 xlabel('s'); ylabel('kPa');
19 subplot(312); axis([0 1000 0 50]);
20 grid on; hold on;
21 xlabel('s'); ylabel('deg C');
22 subplot(313); axis([0 1000 0 100]);
23 grid on; hold on;
24 xlabel('s'); ylabel('kW');
25
26 %Execute chiller for 1000s
27 %
28 %Update water side conditions every 10s
29 t = 10;
30 %Initialize counter and output-storage
31 i = 1;
32 output = [];
33 %Begin loop...
34 while(i<1000)
35     Tewi = TEWI;
36     Tewo_set = TEWO_SET;
37     mewat = MEWAT;
38     mcwat = MCWAT;
39     if(i<120)
40         Tcwi = 21 + i*(TCWI-21)/120;
41     else
42         Tcwi = TCWI;
43     end
44     u = [Tewi:Tcwi:Tewo_set;mewat;mcwat];
45     j = 1;
46     while(j<=t)
47         y = chiller(1,u);
48         output = [output;y'];
49         j = j + 1;
50     end
51     subplot(311);
52     plot(i,y(2),'b.',i,y(3),'r. ');
53     subplot(312);
54     plot(i,y(12),'b.',i,y(13),'r. ');
55     subplot(313);
56     plot(i,y(8),'r. ');
57     save output;
58     i = i+10;
59     pause(0.1);
60 end
61 %Save state at the end
62 chiller(2);
Ready
```

Fig. 39: m-code of *Example 1*.

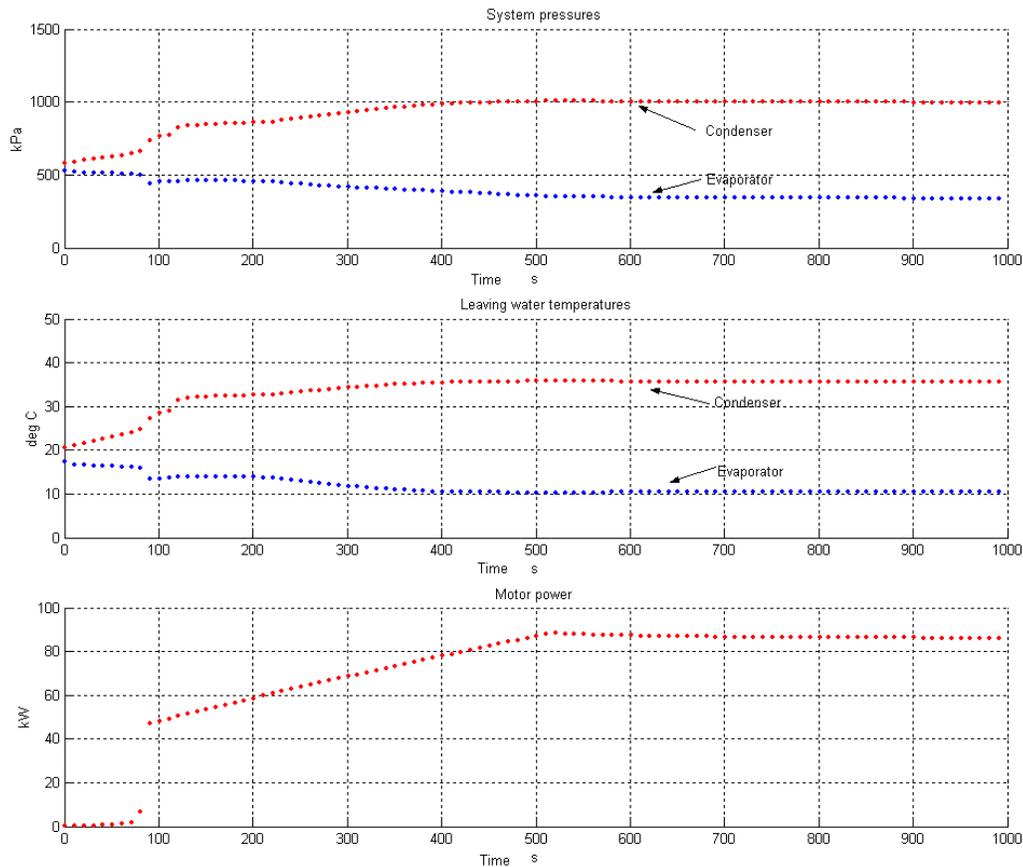


Fig. 40: Output plot of selected chiller parameters in *Example 1*.

During the first 90s of the simulation the model uses a fixed polytropic efficiency for the compressor because the map (eqn. 20) does not apply during that time. This results in a low compressor flow-rate and therefore a low power prediction. This also slows down the early response of the pressures and water temperatures. Once the efficiency map becomes applicable (at 90s), the system's response is seen to change significantly. Thereafter, the solution proceeds smoothly to the steady state. The linear power variation up to ~500s is the effect of the current limit imposed by the controller which prevents the compressor from delivering large flow-rates to rectify the large initial chilled water temperature error.

Example 2 (Ex2.m): Evaporator water entering temperature change in fault-free condition

This example demonstrates the revival of a system state from an earlier saved state⁶, followed by executing the system model by driving it through a transient triggered by a 2°C step drop in the evaporator return water temperature. The step-change occurs 50s after the start of the execution. When steady-state is reached again (150s later), the execution stops and the system state is saved. Fig. 42 shows the m-code for this example. The significant differences are the initialization, which now is done by loading the chiller state from the earlier saved state (line 4), the way the boundary conditions are updated in lines 40-43 and the boundary condition updation and result plotting sampling time (line 29) which is now done every 2s. Fig. 41 shows the output for this example.

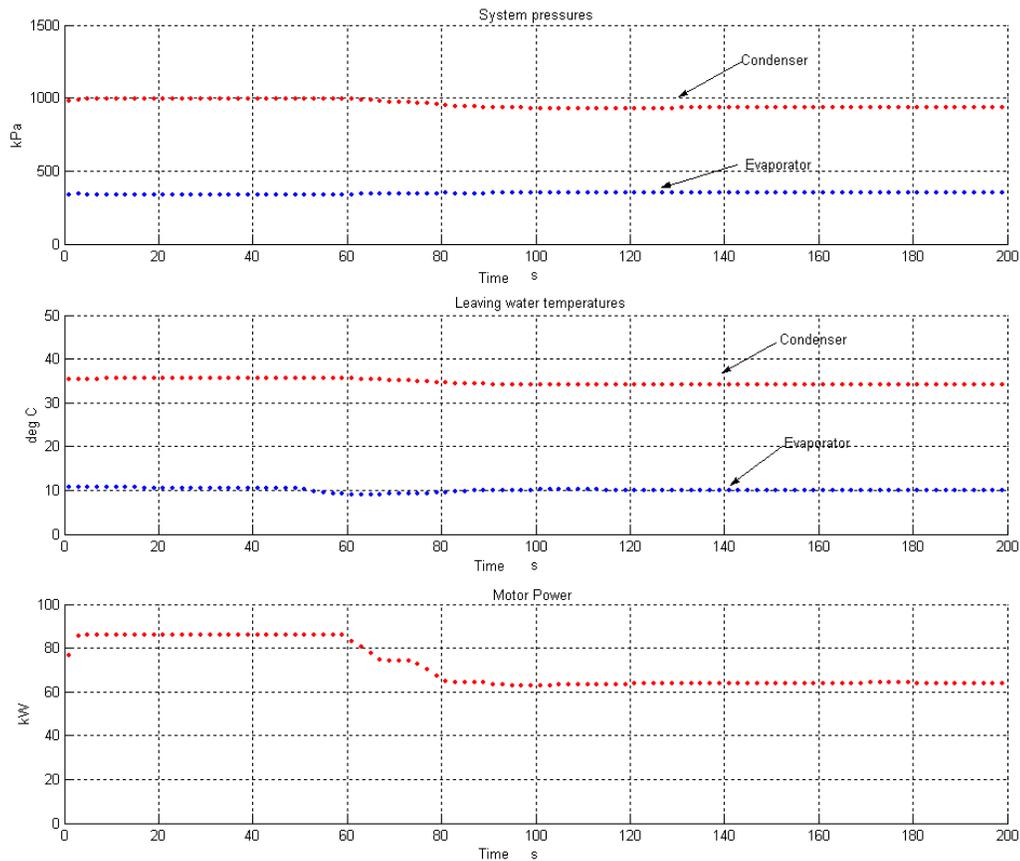


Fig. 41: Output plot of selected chiller parameters in *Example 2*.

⁶ It is assumed here that the text file *SavedState.txt* is as was saved at the end of *Example 1*. If this is not the case, move the *SavedState.txt* file (if one exists) to another location on the disk, copy the file *SavedState_1000.txt* from the *IOFiles* directory to the *Chillersim* directory and rename it *SavedState.txt*.

```

C:\Work\W MatlabInterface\ChillerSim\Ex2.m
File Edit View Text Debug Breakpoints Web Window Help
Stack: Base
1 %Chiller Model Examples
2 %Example 2
3 %Load chiller state
4 chiller(3);
5
6 %Define water-side boundary conditions
7 TEWI = 16;
8 TCWI = 30;
9 TEWO_SET = 10;
10 MEWAT = 13.2;
11 MCWAT = 16.7;
12
13 %Set-up plotting
14 FIG = figure;
15 set(FIG,'Position',[231 132 1128 908]);
16 subplot(311); axis([0 200 0 1500]);
17 grid on; hold on;
18 xlabel('s'); ylabel('kPa');
19 subplot(312); axis([0 200 0 50]);
20 grid on; hold on;
21 xlabel('s'); ylabel('deg C');
22 subplot(313); axis([0 200 0 100]);
23 grid on; hold on;
24 xlabel('s'); ylabel('kW');
25
26 %Execute chiller for 200s
27 %
28 %Update water side conditions every 2s
29 t = 2;
30 %Initialize counter and output-storage
31 i = 1;
32 output = [];
33 %Begin loop...
34 while(i<200)
35     Tawi = TEWI;
36     Tcwi = TCWI;
37     Tewo_set= TEWO_SET;
38     mewat = MEWAT;
39     mcwat = MCWAT;
40     if(i>=50)
41         Tawi = TEWI - 2;
42     end
43     u = [Tawi;Tcwi;Tewo_set;mewat;mcwat];
44     j = 1;
45     while(j<=t)
46         y = chiller(1,u);
47         output = [output;y'];
48         j = j + 1;
49     end
50     subplot(311);
51     plot(i,y(2),'b.',i,y(3),'r. ');
52     subplot(312);
53     plot(i,y(12),'b.',i,y(13),'r. ');
54     subplot(313);
55     plot(i,y(8),'r. ');
56     save output;
57     i = i+t;
58     pause(0.1);
59 end
60 %Save state at the end
61 chiller(2);
Ready

```

Fig. 42: m-code of *Example 2*

A drop in evaporator entering water temperature, keeping the same chilled water set-point and water-flow rate, corresponds to a drop in the building load. This results in lesser heat transfer to the refrigerant in the evaporator. This causes the leaving water temperature to drop below the set-point thus far maintained, as seen in Fig. 41. The reduced evaporator capacity implies that the motor now has to deliver lesser power and the condenser has to reject lesser heat to the cooling water. The stair-step reduction in motor power is caused by the step-and-wait action of the controller that now responds to the negative error in chilled water temperature. The reduced compressor flow rate and necessary condenser heat duty result in the lower condenser pressure and condenser leaving water temperature.

Example 3 (*Ex3.m*): Evaporator and condenser water entering temperature change in fault-free condition:

This example is a repeat of Example 2, with the difference that the condenser entering water temperature is also changed (increased) by 2°C during the transient triggered by a 2°C drop in evaporator return water temperature. The m-code can be seen in *Ex3.m*. Fig. 43 shows the output for this combination transient.

The increased condenser water temperature results in a higher (than in Example 2), condenser pressure and motor power. The higher condenser pressure is caused by the need to sustain the temperature difference between the refrigerant and (the now warmer) water in the condenser which will allow the required heat transfer rate. The stabilized condenser leaving water temperature is also seen to be higher as a result. The motor power is higher because of the increased pressure difference that the compressor now needs to work against.

The condenser leaving water temperature is seen to first drop, in response to the reduced evaporator entering water temperature, and then increase because of the increase in the condenser entering water temperature. The chilled water temperature also responds similarly but the controller manages to return it to the un-changed set-point.

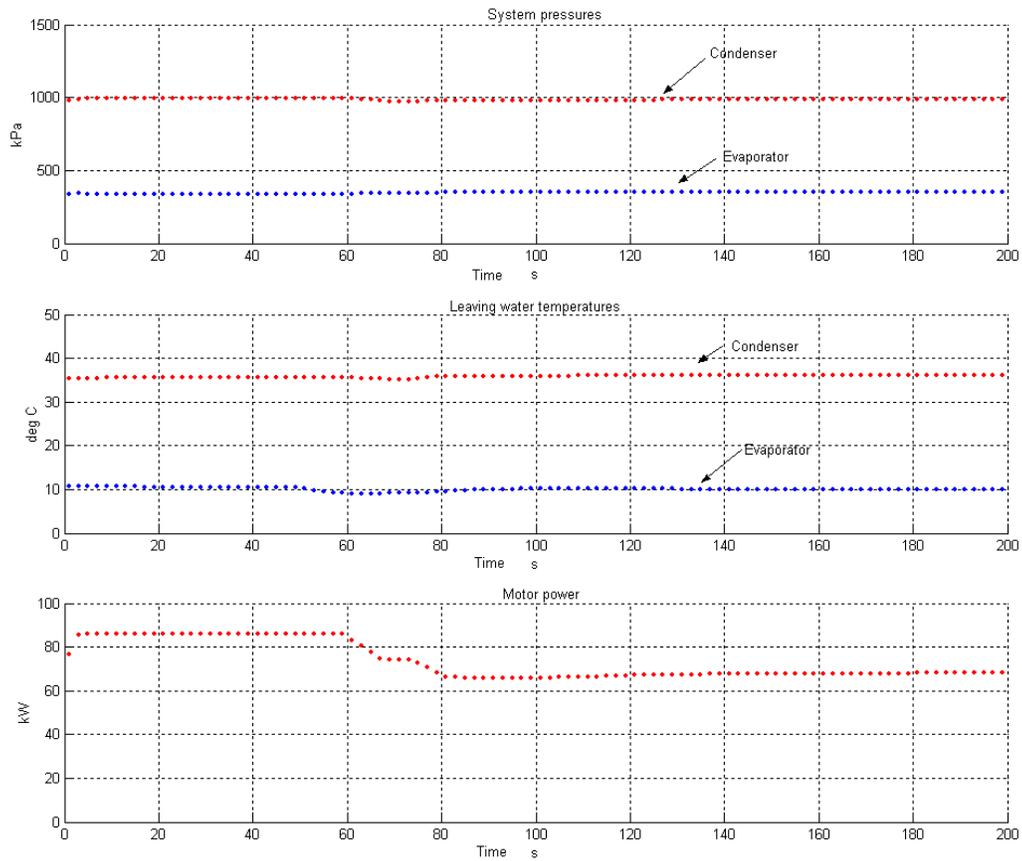


Fig. 43: Output plot of selected chiller parameters in *Example 3*.

Example 4(Ex4.m): Evaporator and condenser water entering temperature change and chilled water set-point temperature change in fault-free condition:

This example includes a 2°C increase in the chilled water set-point temperature into the boundary conditions imposed on Example 3. The system begins with the same initial condition as exists at the end of Example 1 (see footnote on pg. 92). Fig. 44 shows the output. Note the large drop in motor power with the reduced load caused by a smaller evaporator return water temperature as well as increased set point temperature. The chilled water temperature settles down to the new set point stably. The evaporator pressure shows some small scale transients between 100 and 120s. This is caused by the dynamics in the valve.

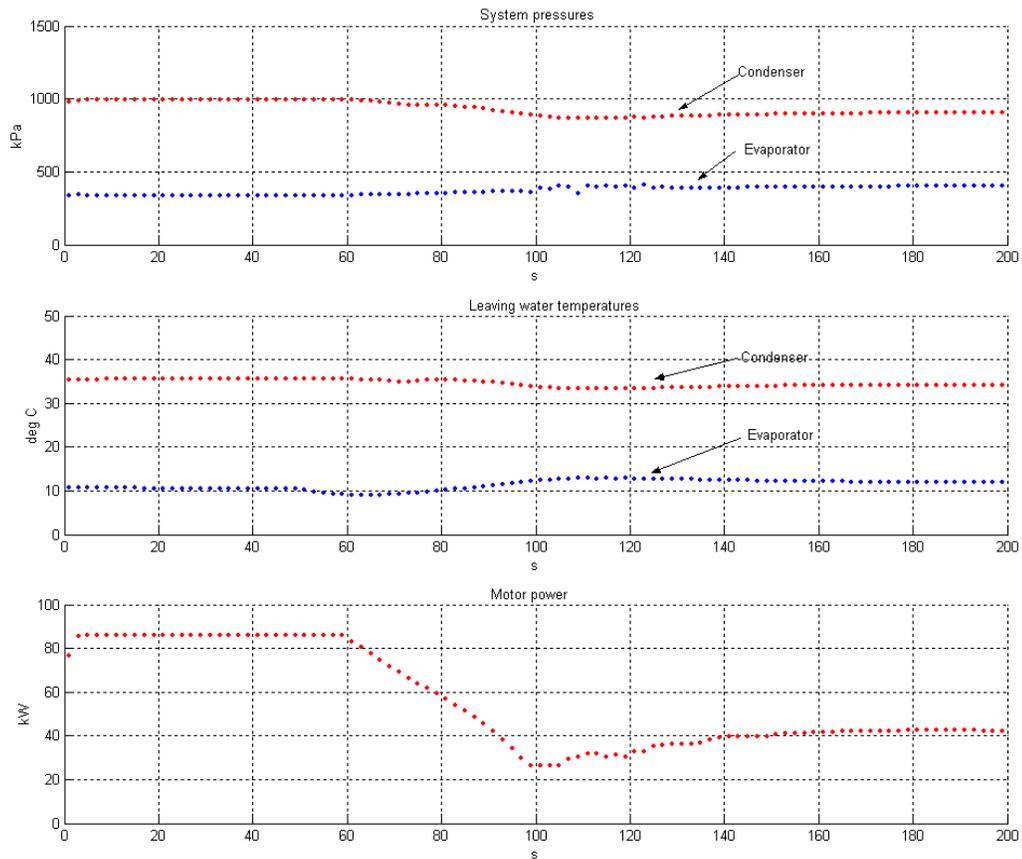


Fig. 44: Output plot of selected chiller parameters in *Example 4*

Fault simulation

The preceding examples demonstrate the use of the model and its output for the fault-free condition. This section demonstrates how the system performance can be generated with faults introduced. The following faults can be emulated in the system model:

- (a) Up to 40% reduction in water flow-rates in one or both heat-exchangers
- (b) Up to 20% refrigerant undercharge in the system
- (c) Up to 20% refrigerant overcharge in the system
- (d) Up to 45% fouling in one or both heat-exchangers

Example 5 (*Ex5.m*) Reduced water flow rates:

This fault can be introduced simply by altering the values entered in the input vector u above. This fault can be introduced either as a fully-developed one or as a gradually developing one. The nominal water flow-rates are 13.2 kg/s in the evaporator water loop and 16.7 kg/s in the condenser water loop. This example demonstrates the reduction in condenser flow-rate as a gradually developing fault. Water flow-rate reduction in the evaporator loop, or in both evaporator and condenser water loops, can also be implemented in the same manner.

The system begins at a state as at the end of Example 1. At this point in time, the condenser water flow rate is at its normal value. Ten seconds after the execution starts the fault begins to develop. Over the next 180s the flow rate drops linearly to 60% of the normal, i.e., 10 kg/s. Fig. 45 shows the output of this simulation.

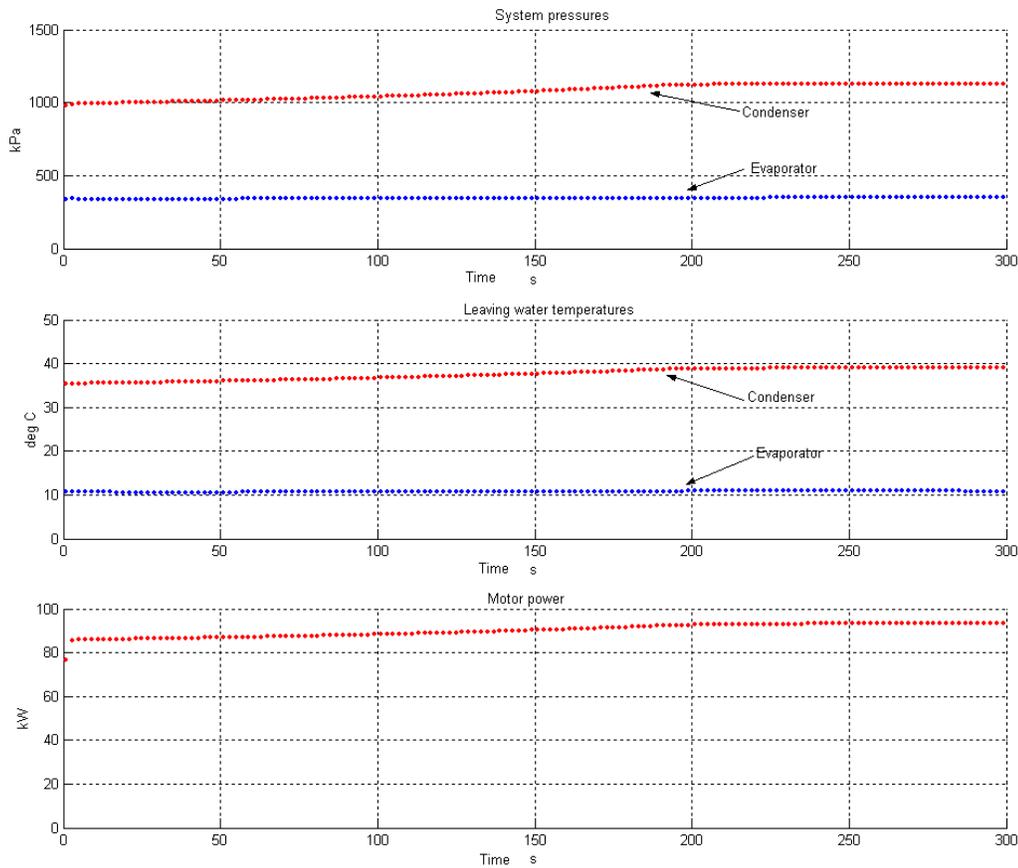


Fig. 45: Output plot of selected chiller parameters in *Example 5*

As expected, the reduced water flow rate results in lesser heat transfer rate on the water side in the condenser. This necessitates a higher temperature difference between the refrigerant and the water, which is achieved by a higher condenser pressure. The evaporator pressure is virtually unchanged. The motor power increases because of the higher pressure difference across the compressor.

Example 6 (Ex6.m): Charge variation:

Refrigerant undercharge and overcharge are implemented in the same manner, i.e., through the initial enthalpy distribution when using a full initialization or by simply specifying the total refrigerant charge when using a minimal initialization. The refrigerant charge quantity normal for the system model is 124.7 kg (see section 4.7). Varying the magnitude of these faults while the model executes is not incorporated in the present code.

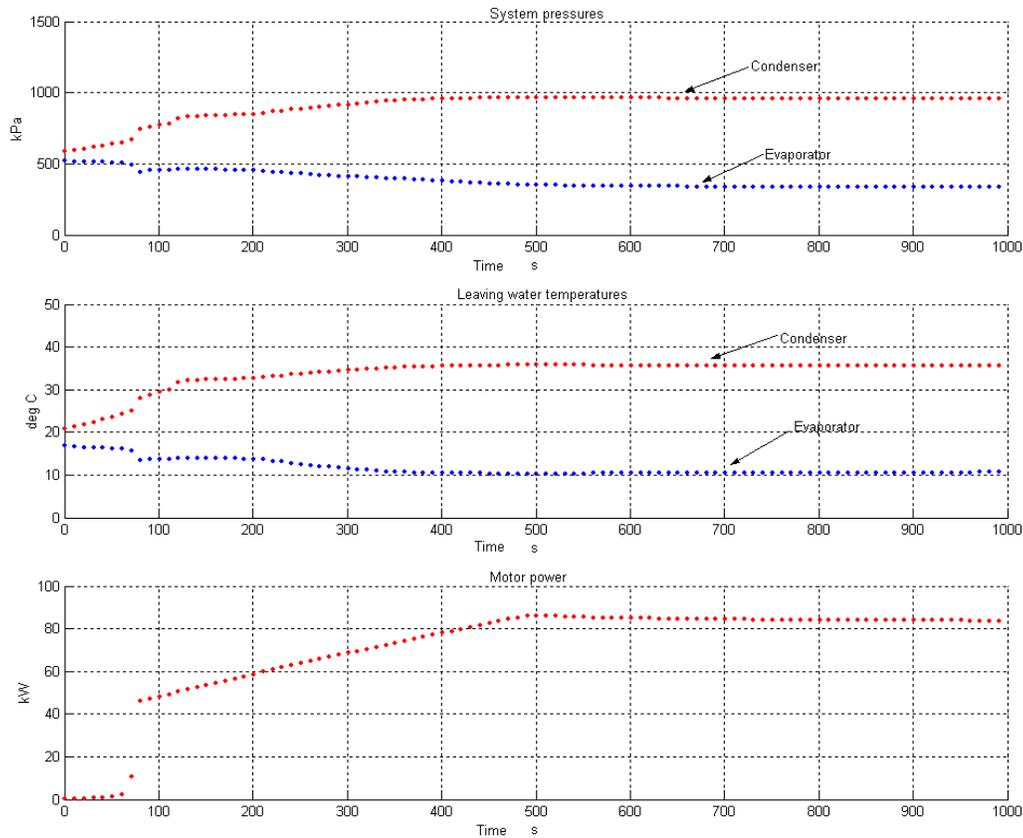


Fig. 46: Output plot of selected chiller parameters in *Example 6*

In this example, the system is initially charged with 80% of the nominal charge, i.e., with 99.76 kg of refrigerant. This is done by altering the third field in *Initial_MINIMAL.txt*. The model is run from start-up through to the achievement of steady-state. The m-code for executing this example is identical to that of *Example 1*. Fig. 46 shows the output. In comparison with the output of *Example 1* which is with the correct refrigerant charge, the system pressures are seen to be lower as is the motor power. Other parameters can be plotted by loading *output.mat* into the Matlab workspace. The columns of *output* are indexed identically to the output vector *y* listed earlier in this section.

Refrigerant overcharge can be similarly implemented by setting the value in *Initial_MINIMAL.txt*. Alternatively, for either case of charge variation, the enthalpy distribution can be set in *Initial_FULL.txt* and the chiller can be initialized by a single-argument call with a value of 1. The refrigerant volumes of the heat exchangers can be determined from their geometry files.

Example 7: Heat exchanger fouling:

Heat exchanger fouling is implemented by specifying fouling as a percentage loss in heat transfer conductance or area. The fouling applies only to the water-side of the tube. This fault parameter is entered along with the heat exchanger geometry, as the sixth field of the second line in the geometry file shown circled in Fig. 47. The magnitude of fouling cannot be altered during model execution in the present code. Any of the example m-codes can be used with this change made in the geometry file.

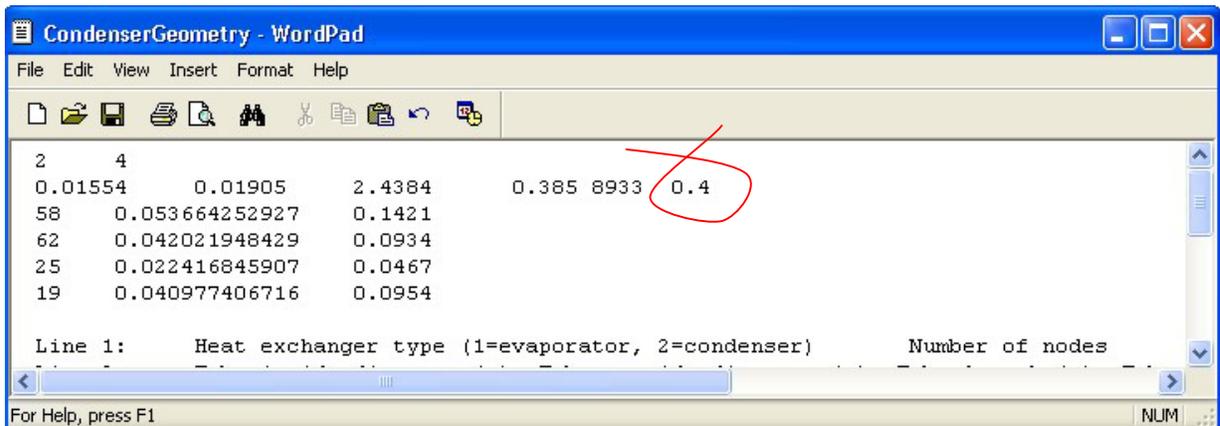


Fig. 47: Screen-shot of condenser geometry file with 40% fouling introduced

Example 8 (Ex8.m): Boundary condition updation from file:

The final example m-file is the file used to execute the chiller model through the data presented under the section on validation. This file demonstrates how the boundary conditions stored in a text file on disk can be read at pre-determined intervals and used to execute the chiller model. The user is encouraged to examine the code in this m-file as much of it is self-explanatory.

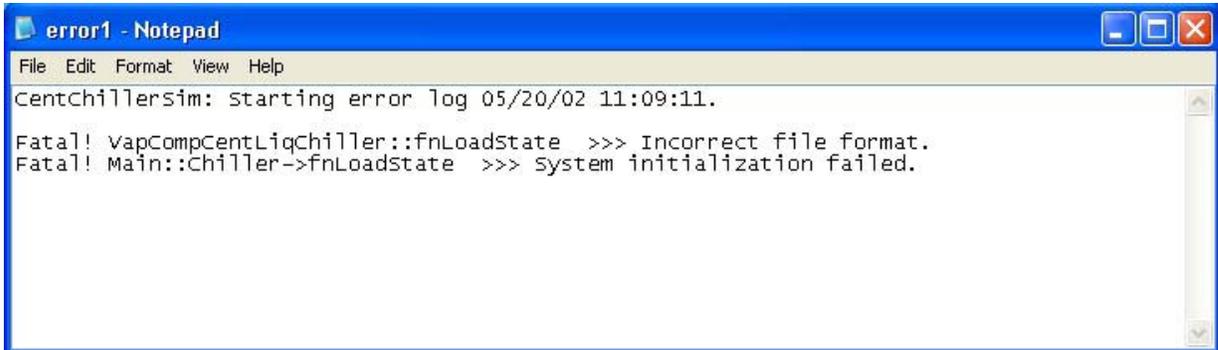
Error handling and reporting

Error messages generated are broadly divided into those that are launched from within the Matlab interface and those that are launched from within the model code. The errors trapped within the model code are logged to a text file *error1.log*, which is created automatically when the chiller model is run for the first time in any session. The Matlab interface generated errors consist of the following:

1. 'Invalid call to chiller routine' – the number of arguments passed and returned in the call to chiller does not match either the single-argument call format or the two-argument call format described above.
2. 'Cannot create engine' – a chiller model could not be created in memory, likely because of insufficient memory.
3. 'System construction failed' – defining the system components geometry failed. This can occur if the paths to the geometry files have not been included into Matlab's search path or if any of the geometry files are of incorrect format. In case of the latter, additional error information is logged to *error1.log*.
4. 'Initialization code needs to be an integer' – the value passed in a single-argument call to chiller was not an integer. Passing anything other than an integer will trigger this error.
5. 'Invalid argument value' – the value passed in a single-argument call to chiller was not any of 0,1,2 or 3. These are the only acceptable values for a single-argument call.
6. 'System state could not be saved' – saving the system state in the text file *SavedState.txt* failed. A possible cause of this is insufficient disk space.

7. 'System state could not be loaded' – loading a previously saved system state failed.
Possible causes of this are:
 - a. the file *SavedState.txt* was not found because the path was not included into Matlab's search path
 - b. the format of information in the file *SavedState.txt* is incorrect. Additional error information is logged in *error1.log*.
 - c. one or more of the system components could not be initialized with the data in *SavedState.txt*. Additional error information is logged in *error1.log*.
8. 'System initialization failed' – the system components could not be initialized with the data in *Initial_FULLL.txt*. Possible causes are:
 - a. the file *Initial_FULLL.txt* was not found because the path was not included into Matlab's search path
 - b. the format of information in the file *Initial_FULLL.txt* is incorrect. Additional error information is logged in *error1.log*.
 - c. one or more of the system components could not be initialized with the data in *Initial_FULLL.txt*. Additional error information is logged in *error1.log*.
9. 'Chiller not initialized' – an attempt was made to use the chiller model without performing an initialization.
10. 'Chillersim error. Consult error log file for details' – an error occurred in the model code and additional error information has been logged in *error1.log*.

Fig. 48 shows an example of error information that is logged into *error1.log*. The error entry shown is that of an error in the format of information provided in the initialization file. The error messages are logged from local to global, i.e. the first error message identifies the location in the code where the effect of the error was observed. Subsequent error messages help identify the path through the code that was being executed at the time the error occurred. Since the cause of the error can lie away from the point where its effect is observed, the error message can only be generic and a full listing of these error messages serves little purpose.



```
error1 - Notepad
File Edit Format View Help
CentChillersim: starting error log 05/20/02 11:09:11.
Fatal! VapCompCentLiqChiller::fnLoadState >>> Incorrect file format.
Fatal! Main::Chiller->fnLoadState >>> system initialization failed.
```

Fig. 48: Screen-shot of error log file with an example error entry

The error trapping structure has been designed to trap most of the common errors that were conceived as possible. As usage of the model increases, additional information will be available from users that will help identify errors and bugs not yet detected and the code can be made more robust.